Research Article

Advertising Click-Through Rate Prediction Model Based on an Attention Mechanism and a Neural Network

Lei Lei, Ziqi Jia, and Yangxue Dong

School of Computer and Software, Nanyang Institute of Technology, Nanyang 473000, China

Correspondence should be addressed to Lei Lei; 3162098@nyist.edu.cn

Received 8 June 2022; Revised 3 August 2022; Accepted 10 August 2022; Published 29 August 2022

Copyright © 2022 Lei Lei et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the rapid development of the Internet and the rapid expansion of the advertising market, display advertising has become the most popular means of publicity. Accurate advertising recommendation is the guarantee of Internet platform revenue, and accurate advertising click-through rate prediction is the premise of accurate recommendation. According to the requirements of the update rate and real-time of the advertising platform, the coupling relationship of the advertising business module can be divided into two categories: offline business module and online business module. The offline advertising management module is mainly based on the construction of the mathematical model, which is used to mine the complex relationship between users and commodity characteristics; the online advertising management module is mainly based on the real-time feedback of users and changes the recommendation strategy in real time by collecting feedback information. For the offline advertising management module, an advertising click-through rate prediction model based on attention mechanism and neural network is proposed, which is called CAN. This method provides richer feature interaction information for a neural network layer. For the online advertising management module, a real-time recommendation algorithm based on Gaussian process is proposed. This method can solve the universality of specific function assumptions in various environments. Through the coupling of the advertising click-through rate prediction model constructed by the offline advertising management module and the real-time recommendation algorithm of the online advertising management module, the accurate delivery of advertising can be realized.

1. Introduction

In the age when the Internet is not popular, advertisements are mainly spread by traditional news media, and the main ways of communication are TV, roadside billboards, newspapers, radio, and so on [1]. These traditional advertising display methods need a lot of manpower to publicize, where the input cost is high and the income is slow, and it is difficult to manage effectively. Internet advertisement refers to the advertisement put on multimedia or network, which came into being in 1995. Its appearance changed the traditional way of propaganda, and advertising service providers began to display their products with browser web pages as the main propaganda media [2]. With the development of mobile terminals, Internet advertisements appear more in major applications. Compared with traditional media, online advertising has wide coverage, strong pertinence, and large audience base. The network provides a new and efficient connection mode for consumers and enterprises, which makes online advertising flexible and interactive. Enterprises can immediately collect consumer feedback data through the network, such as search query records, advertisement click records, problem feedback, and other information. Therefore, the coupling relationship model between the offline advertising management module and the online advertising management module based on the machine learning model proposed in this paper can quantitatively analyze the collected data platform and further propose improvement schemes, such as refining delivery targets, changing delivery methods, and adjusting budget allocation strategies. With the wide application of the deep neural network, the traditional click-through rate and the conversion rate prediction model based on the machine learning algorithm is gradually replaced by the deep learning...
model [3]. The model of deep neural network can extract the user’s interest characteristics and delay relationship from multisource information and predict the user’s future behavior and advertising effect [4]. It summarizes the feature learning and technology research of online advertising click-through rate prediction and analyzes the research status of online advertising click-through rate prediction at home and abroad from the aspects of original data characteristics and solutions, feature learning of click-through rate prediction, click-through rate prediction model construction, evaluation index selection, and so on [5]. Click-through rate estimation can be applied to Internet advertising, recommendation system, and other fields, and has a high research value [6].

2. Machine Learning Model

2.1. Feature Classification. Both offline recommendation and online recommendation systems pay more and more attention to feature learning. Features contain the attributes of goods and users. Making full use of feature information can greatly improve recommendation efficiency [7]. From the perspective of machine learning, features can be divided into continuous features and discrete features.

(1) Continuous characteristics: The eigenvalue is a continuous value.

(2) Discrete features: The values of features are fixed type attributes. Usually, discrete features can give an attribute list, and the values of specific features can be one or more values in the list.

Logistic regression algorithm in machine learning has a large number of applications in advertising recommendation industrialization, which regards each feature as an independent attribute, and each attribute is given different weights. This approach thinks that features are independent, thus ignoring the correlation between features. In the latest algorithm theory, cross feature is also called feature combination. As a new concept, it is considered the key to further improve the accuracy of recommendation. According to the complexity of cross features, feature crossover is divided into low-order cross features and high-order cross features, and a cross feature composed of n features is defined as n-order cross features. The combination of first-order and second-order features is defined as low-order cross features, and the combination of features above third-order is defined as high-order cross features.

From the method of feature combination, it includes the following two schemes [8]:

(1) Artificial feature combination: Feature crossover is mainly carried out by experts selecting features according to experience.

(2) Automatic feature combination: The corresponding combined features are automatically generated by algorithms.

Automatic feature combination can be divided into two categories:

(1) Explicit feature combination: Select the features to be combined and use heuristic search to find the best feature combination. This method defines the features to be combined in advance and has strong interpretability. The combined features obtained can be used as the training basis of other machine learning algorithms. The disadvantage is that it is difficult to find the optimal feature combination solution, which requires a lot of computing space to support it.

(2) Implicit feature combination: It mainly combines features through a neural network, which is very suitable for processing continuous features, such as images and speech. Due to the use of a deep neural network, it is almost not explanatory.

2.2. Offline Advertising Management Module. The offline recommendation system mainly transforms the recommendation problem into the prediction problem of the advertisement click-through rate. The algorithm mainly relies on building a specific mathematical model and training with large data sets to optimize the model parameters [9]. The model accurately predicts the click-through rate of users to recommend advertisements, so as to recommend the best advertisements to users. Traditional offline advertising recommendation algorithms describe the relationship between users and products by constructing low-order cross features, which is called a low-order click-through rate prediction model. With the vigorous development of deep learning, more and more research studies combine the deep neural network with an offline recommendation model to construct rich nonlinear implicit features, which further improves the prediction accuracy and is called a high-order click-through rate prediction model.

This section introduces some algorithms of low-order and high-order click-through rate prediction models. Low-order prediction models include logistic regression, factorization machine, and field perception factorization machine. High-order prediction models include Wide&Deep, DeepFM (Deep Factorization Machine), NFM (Neural Factorization Machine), and PNN (Product Neural Network).

2.2.1. Logistic Regression. The most classical logistic regression algorithm in machine learning is also widely used in advertising click-through rate prediction [10]. Firstly, the algorithm assumes that commodity x has k features, which are expressed as $x = (x_1, x_2, \ldots, x_k)$, and $x_i$ represents the $i$th feature value of commodity $x$. The relationship between features and user feedback is defined by a first-order linear model, as shown in the following formula:

$$f(x) = \sum_i w_i x_i + b.$$  \hspace{1cm} (1)

In vector form, we get

$$f(x) = w^T x + b.$$  \hspace{1cm} (2)
The feedback result \( f(x) \) is converted to the user click rate by the sigmoid function, which is shown in the following formula:

\[
y = \frac{e^{w^T x + b}}{1 + e^{w^T x + b}}
\]  

(3)

The parameters \( w \) and \( b \) are calculated by the maximum likelihood estimation in probability theory. Given the data set, the maximum logarithmic likelihood is

\[
I(w, b) = \sum_{i=1}^{k} \ln p(y_i | x_i; w, b).
\]  

(4)

Let \( \beta = (w; b), x = (x; 1) \), then \( w^Tx + b \) can be abbreviated as \( \beta^Tx \). Then, make \( p_1(x; \beta) = p(y = 1 | x; \beta), \) \( p_0(x; \beta) = p(y = 0 | x; \beta) = 1 - p_1(x; \beta) \), then formula (4) can be written as

\[
p(y_i | x_i; w, b) = y_i p_1(x_i; \beta) + (1 - y_i) p_0(x_i; \beta).
\]  

(5)

\[
I(\beta) = \sum_{i=1}^{k} \left( -y_i \beta^T x_i + \ln(1 + e^{\beta^T x_i}) \right).
\]  

(6)

Formula (6) is a higher-order differentiable continuous convex function about \( \beta \). According to the optimization theory, the optimal solution can be obtained by the steepest descent method and the classical Newton method. LR (Logistic Regression) algorithm has fewer parameters, simple model, easy iteration, and relatively easy parameter adjustment. However, the LR algorithm is extremely dependent on the quality of feature selection, and the cost is high. With the increase of the amount of feature information, the original feature dimension increases rapidly, and the number of features increases sharply. It is difficult to extract information accurately by the manual method. Moreover, this method does not consider the influence of cross features, so the prediction ability of the model is limited.

2.2.2. Factorization Machine. Factorization machine (FM) is a new model proposed by Rendle [11] in 2010, which combines a matrix factorization model and a support vector machine. It focuses on the influence of second-order cross features on prediction accuracy. According to the idea of the logistic regression algorithm, the simplest second-order crossover method is shown in formula (7). However, the features of actual commodities are mostly discrete features. After single thermal coding, the data dimension is large, the effective data is extremely sparse, and most of the feature values are 0. At this time, for the simple implementation of formula (7), the cross product term \( x_ix_j \) of most features is 0, so this implementation cannot be directly trained.

\[
y = w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} w_{ij} x_i x_j.
\]  

(7)

The factorization machine performs matrix factorization on the weight factor \( w_{ij} \) and sets two hidden vector factor matrices \( v \) and \( v^T \) constructs \( w_{ij} \) by multiplying them. The hidden vector factor matrix \( v \) is expressed as follows:

\[
y = \left( \begin{array}{c} v_1 \cdots v_{1k} \\ \vdots \cdots \vdots \\ v_n \end{array} \right) = \left( \begin{array}{c} v_1 \\ \vdots \\ v_n \end{array} \right) = \left( \begin{array}{c} v_{11}^T v_{12}^T \cdots v_{1k}^T \\ \vdots \cdots \vdots \\ v_{n1}^T v_{n2}^T \cdots v_{nk}^T \end{array} \right).
\]  

(8)

where \( v_i \) is the ith hidden factor vector, \( k \) is the dimension of the hidden factor vector, and the formula for calculating the weight \( w_{ij} \) of the second-order cross feature \( x_ix_j \) is shown in the following formula:

\[
w_{ij} = v_i^T v_j = \left( \begin{array}{c} v_{11} \\ \vdots \\ v_{nk} \end{array} \right) \left( \begin{array}{c} v_{1i} \\ \vdots \\ v_{nj} \end{array} \right) \left( \begin{array}{c} v_{i1}^T v_{i2}^T \cdots v_{ik}^T \\ \vdots \cdots \vdots \\ v_{n1}^T v_{n2}^T \cdots v_{nk}^T \end{array} \right).
\]  

(9)

Combining formula (9), the prediction function of the factorization machine is expressed as shown in the following formula:

\[
y = w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} v_i^T v_j x_i x_j.
\]  

(10)

Factorization machine algorithm can automatically construct the second-order cross feature by implicitly the vector inner product, and the time complexity is \( O(nk) \). Combined with logistic regression algorithm, it can complete the prediction in linear time, which is simple and efficient.

2.2.3. Field Sensing Factorization Machine. Field-aware factorization machine is called the field-aware factorization machine (FFM), which is further improved on the basis of the factorization machine. The algorithm considers that hidden vectors are not only related to features but also related to the field to which features belong. Feature domain is defined as the classification of similar features. Take the attributes of clothes as an example: clothing softness, material, and other features can be classified as material domain, and clothing style and color can be classified as appearance domain. The intersection of each feature and other features is further refined into the intersection relationship between features and other classes in the FFM algorithm. FM considers a feature to be a single domain, so the FM algorithm can be regarded as a special case of the FFM algorithm.

The prediction function of the FFM algorithm is shown in the following formula:

\[
y(x) = w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \langle v_{i,f} \rangle v_{i,f}^T x_i x_j.
\]  

(11)

Compared with FM, the number of parameters to be learned by a single feature \( x_i \) is extended to \( f \times k, f \) is the
number of domains, $k$ is a single hidden factor dimension, and the overall time complexity is $O(kn^2)$. FFM algorithm has a large number of parameters and cannot be optimized into a linear model, so it has a large amount of calculation and low efficiency under large-scale data.

2.2.4. Wide&Deep. Wide&Deep model is an advertisement click-through rate algorithm proposed by Google in 2016 and applied to GoogleStore [12]. The algorithm puts forward two concepts: memorization and generalization. The memory of the model is defined as the system discovers the known merchandise and user-related characteristics from the user browsing information recorded in the log. The generalization of a model is defined as the ability of a system to obtain little or no feature correlation in historical data. In the Wide&Deep algorithm, the memory is reflected in the algorithm using expert system to manually select features that have great influence on prediction results according to experience, while the generalization is reflected in the model using a deep neural network to mine hidden feature combinations.

However, the input of the two parts of the Wide&Deep model is different, and the Wide part needs to be selected manually, so the algorithm is not an end-to-end model, and there are some shortcomings in dealing with large-scale datasets. However, Wide&Deep lays a foundation for how to combine the neural network with advertisement click-through rate prediction.

2.2.5. Neural Factorization Machine (NFM). Wide&Deep and DeepFM are typical parallel prediction models. Linear model and neural network model are trained at the same time, and the prediction results are given together with the results of the two models. Dr. He Xiangnan and others proposed that this parallel structure will make model training difficult and proposed a serial model neural factorization machine, which organically combines the FM linear model and the neural network nonlinear model in high-order feature crossover. The model diagram is shown in Figure 1.

NFM algorithm proposes a series layer bi-interaction before the higher-order neural network layer. This layer combines any two input features to construct a new vector and then adds all vectors as inputs of the neural network layer, as shown in the following formula:

$$f_{BI}(V_x) = \sum_{i=1}^{n} \sum_{j=i+1}^{n} x_i \cdot v_i \odot x_j \cdot v_j = \frac{1}{2} \left[ \left( \sum_{i=1}^{n} x_i \cdot v_i \right)^2 - \sum_{j=1}^{n} (x_j \cdot v_j)^2 \right],$$

(12)

where $o$ is defined as the bit calculation of vector, so it is assumed that the dimension of input vector is $k$, and the output of bi-interaction is $k$. The bi-interaction layer is followed by the high-order feature interaction constructed by the neural network layer [13]. Compared with DeepFM and Wide&Deep algorithms, the input dimension of the DNN layer is greatly reduced, and the time complexity is $O(kn)$, which makes the model complete the calculation in linear time, and the training difficulty is less than the previous algorithms.

2.2.6. PNN. The PNN algorithm is proposed for the insufficient cross-combination representation of the original feature input into the neural network [14], which further improves the bi-interaction layer in NFM, as shown in Figure 2.

For the sake of description, this paper assumes that the commodity feature set $x = \{x_1, x_2, \ldots, x_n\}$, where $x_i = \{x_{i1}, x_{i2}, \ldots, x_{ik}\}$, and $k$ represents the dimension of each feature. The $Z$ function portion connects the original features directly in series as an input feature, as shown in the following formula:

$$z = contact([x_1, x_2, \ldots, x_n], axis = 1).$$

(13)

The $P$ part uses the form of vector inner product to express the feature interaction relationship and multiplies the original features pairwise to obtain a vector $p$ with a length of $n \times (n-1)/2$, as shown in the following formula:

$$p = contact([x_1 x_2, x_1 x_3, \ldots, x_i x_j], axis = 1),$$

$$i = j, \quad j = 1, 2, \ldots, n.$$

(14)

The product layer joins the vector $P$ and the series vector $Z$ as the input of the neural network layer and trains with the dataset. Compared with NFM, a PNN algorithm increases the output attributes of a low-order linear interaction layer, which contains both original features and interaction feature information and further alleviates the training difficulty of neural network.
2.3. Online Advertising Management Module. Mandai et al. improved the UCB algorithm, modeled online recommendation as a context-based gambling machine problem, and proposed a dynamic recommendation algorithm using serialization features of items to make decisions [15]. In the context scenario, the algorithm makes decisions according to the characteristics of the items to be recommended and adjusts the overall decision strategy based on the user’s click feedback to maximize the user’s click effect. For newly generated items, the algorithm can process their features immediately and then add them to the recommendation decision sequence.

LinUCB algorithm assumes that each item $x$ corresponds to a $d$-dimensional eigenvector $x \in \mathbb{R}^d$, and assumes that there is a linear relationship between the expected reward $r_x$ of an item and its eigenvector, which can be expressed by the coefficient vector $\theta_x \ast \in \mathbb{R}_d$ corresponding to item $x$. This paper assumes that the eigenvector of item $x$ at time $t$ is $x_t$ and the reward at time $t$ is $r_{x,t}$, then the expected reward satisfies the following formula:

$$E[r_{x,t}|x_t] = \theta_x^T x_t. \quad (15)$$

Although LinUCB builds a learning model with projects as the main body, it builds a model $\theta_x$ for each project $x$ and modifies the parameter $\theta_x$ by constantly interacting with user feedback information. However, the features observed by the algorithm each time are actually the user features observed from the perspective of the project. After that, for each item to be recommended, calculate the expected reward and the upper bound of confidence interval that the target user can get and select the appropriate item to recommend to the user from two aspects.

3. Analysis of Coupling Relationship of Advertising Management Modules

3.1. CTR Model Based on the Attention Mechanism and the Neural Network. The common practice of offline advertising management module is to convert the recommendation problem into the prediction problem of user click-through rate CTR and improve the recommendation effect by displaying advertisements with the high prediction click-through rate.

CAN algorithm deeply depicts the complex relationship between users’ behavior of clicking advertisements and features and explores the low-order and high-order interaction between features through the combination of the attention mechanism and the deep neural network. The model framework is shown in Figure 3. The proposed algorithm firstly uses embedding technology to compress and reduce the dimension of the original encoded features. After that, the attention mechanism is used to construct low-order cross features, and then the tensor containing low-order interaction information is injected into the deep neural network to mine the high-order nonlinear interaction. Finally, the nonlinear feature tensor is converted into the corresponding predicted advertisement click probability through the prediction layer. The rest of this section introduces the whole algorithm in modules.

3.1.1. Input and the Embedding Layer. Each discrete feature is composed of a large number of zeros and ones after single or multi-thermal coding, while continuous features are composed of continuous values. CAN algorithm can deal with these two kinds of features at the same time.

3.1.2. Interaction Layer of the Attention Mechanism. The input of the attention mechanism interaction layer is the feature of the embedding layer after dimension reduction. The attention mechanism interaction layer carries out second-order cross modeling on input features and reconstructs new feature vectors according to the attention weight values between features. The principle is shown in Figure 4.

From the dimension point of view, the dimension of $e_n$ is the same as that of the original feature input, but the vector reconstructed by attention mechanism contains related information with other features.

$$e_m = \sum_{k=1}^{n} a_{mk} \ast e_k. \quad (16)$$

3.1.3. Neural Network Layer. The neural network layer uses fully connected neural network to mine the high-order nonlinear relationship between features.

The input of this layer is the output vector $Z$ reconstructed by the attention mechanism interaction layer, and the weights are multiplied between neurons in each layer. The calculation between neurons is shown in the following formula:

$$z_{i+1} = f(W_i z_i + b_{i+1}). \quad (17)$$

3.1.4. Prediction Layer. The prediction layer integrates the output of the neural network layer and predicts the user’s click-through rate according to the corresponding tensor,
which shows the effect of the recommendation system intuitively. The calculation process is as follows:

\[ p = \text{Sigmoid}(h^T y), \]

\[ \text{Sigmoid}(x) = \frac{1}{1 + e^{(-x)}}, \]  \hspace{1cm} (18)

The predicted results are compressed by sigmoid function. Sigmoid’s function is shown in Figure 5, and the prediction result is mapped to a number between 0 and 1, which indicates the actual meaning of the coincidence probability value between 0 and 1.

3.2. Dynamic Recommendation Model Based on Gaussian Process. Based on Gaussian process, two online advertising management models are proposed. Both methods can adjust the recommendation strategy online according to the real-time feedback of users and get the favorite products of users with fewer attempts. In addition, the two proposed methods are context-dependent, that is, the online recommendation algorithm combining the characteristics of users and items and can recommend dynamically changing commodity pools immediately.

3.2.1. Gaussian Process. Gaussian process is a widely used mathematical model, which is often used to express the distribution of functions. Different from the traditional model-based machine learning method, Gaussian process does not express the model by parameters but uses distribution to express the corresponding function. The advantage of the parametric representation method is that it can fit all black box functions, which makes the model no longer stick to the adjustment of super parameters, and can efficiently simulate the uncertainty of functions. For the quantification of uncertainty, limited training data can be used to explore the data areas that are least likely to achieve efficient training.

3.2.2. Algorithm Design. This section first gives the recommendation flow chart of the online recommendation system, as shown in Figure 6.

The commodity platform selects a certain number of commodities from the content library as the commodity candidate pool \( I \). In the \( t \)-round recommendation, the recommendation system recommends the product \( I_t \) to the user according to the recommendation strategy of the corresponding preapplication training model, and the user makes corresponding feedback to the product. After each round, the system adjusts the recommendation strategy in time according to real-time feedback from users.

4. Experimental Simulation Analysis

4.1. Evaluation Index. In this paper, LogLoss and AUC are used to analyze the coupling relationship of advertising management modules. The two indicators have different concerns about the recommendation effect. LogLoss directly reflects the difference between the predicted value and the real value through cross information entropy, while AUC pays more attention to the rationality of ranking recommendation results.

(1) LogLoss: This index mainly judges the accuracy of recommendation based on the difference between the predicted value and the actual value. The smaller the LogLoss value, the closer the recommended result is to the real evaluation of users, and the better the prediction effect of the system. This index is used as the loss function of CAN and contrast model training. The evaluation criteria are shown in the following formula:

\[ \text{Logloss} = \frac{1}{N} \sum_{i=1}^{N} y_i \log(P_i) + (1 - y_i) \log(1 - P_i), \]  \hspace{1cm} (19)

where \( P_i \) is the click-through rate of the \( i \)-th advertisement; \( y_i \) is the real clicks of advertising users; \( N \) is total number of ads in the test set.

(2) AUC: The AUC value is equal to the probability that the ranking of randomly selected positive samples is higher than that of randomly selected negative samples. The higher the AUC, the better the coupling performance of the advertising management module.
In the actual simulation, formula (20) is used to calculate the AUC value:

\[
AUC = \frac{\sum_{i=1, j=1}^{M, j=N} \delta (r_i - r_j > 0) + 0.5\delta (r_i - r_j = 0)}{M \times N},
\]  

(20)

where \(M\) represents the number of positive samples; \(N\) represents the number of negative samples; \(r_i\) is the predicted score of the positive sample; \(r_j\) is the predicted score of negative samples; Delta\(\delta(x)\) is an indicator function; \(x\) is a Boolean variable. When \(x\) is true, \(\delta(x)\) is 1, and conversely, \(\delta(x)\) is 0.

4.2. Comparison of Prediction Accuracy. Under the two verification criteria, CAN algorithm performs better. CAN, NFM, and PNN are all algorithms that combine DNN and shallow interaction model, and LogLoss and AUC values are obviously improved compared with the DNN algorithm. This proves that adding a low-order feature interaction model in front of the neural network layer and reconstructing the original input features can effectively alleviate the training difficulty of the network and improve the prediction accuracy. It is shown in Figure 7.

In addition, the three schemes are better than the FM algorithm in two indexes. This verifies that simple linear feature crossover has insufficient ability to characterize the complex relationship between users and commodities, and the deep neural network can mine more complex nonlinear interactive relationships. Comparing these three algorithms, the prediction effect of CAN is further improved than that of PNN and NFM. This shows that using the attention mechanism to construct low-order feature crossover is more effective than PNN and the NFM algorithm to simply multiply the original features. In addition, in the simulation results, it is found that the prediction effect of DNN is not as good as that of the FM algorithm. For this phenomenon, it is guessed that MovieLens data has small collective quantity, few features, and simple interaction relationship, and better results can be achieved by simple second-order linear crossover, while a complex neural network makes the model complex and increases the calculation overhead and reduces the prediction accuracy.

In order to verify the above conjecture, the same simulation experiments are carried out on Criteo datasets with larger data and features. The result is shown in Figure 8.

For Criteo datasets, the prediction results of the FM model are obviously worse than those of other models combined with neural networks. This result verifies that for data with multiple feature types and highly sparse data, the model that is using only low-order feature crossover cannot...
represent complex feature relationships. In addition, in this experiment, the effect of the NFM algorithm is slightly worse than the DNN algorithm, which shows that simple multiplication of features in front of the neural network layer plays a negative role. Compared with DNN and PNN, the effect of CAN is improved to some extent, which means that in an extremely complex feature environment, it is more effective to use the attention mechanism to construct low-order feature interaction than to use simple value multiplication in NFM and PNN.

4.3. Low-Order Interaction. As can be seen from Figure 9, with the continuous training of the model on the training set, the AUC values of the two algorithms are continuously improved on the training set. However, in the verification set, after the DNN algorithm obtained the best prediction effect in the 13th round, the AUC value decreased rapidly, and the downward trend gradually flattened after the 30th round. Although the CAN algorithm has the same phenomenon, with the increase of training rounds, AUC on the verification set gradually decreases. However, compared with the DNN algorithm, the decline rate is greatly reduced, which shows that compared with the original features, injecting feature vectors containing interactive information into the neural network layer is indeed helpful to reduce the training difficulty and the burden of mining interactive information on the network to a certain extent, thus alleviating the over-fitting phenomenon.

4.4. Hyperparametric Influence. In this section, this paper will study the embedding layer output size, the number of neural network layers, and the number of neurons in each layer on the recommendation effect. All experiments are based on the MovieLens dataset, and the experimental parameters of the two indicators are set the same.

The setting of the fixed dimension size of the embedding layer (named as the embedding size) affects the output size of subsequent models. This paper observes the change of the
prediction effect by changing the embedding size, and the result is shown in Figure 10.

In the LogLoss experiment, with the increase of the embedding size, the LogLoss decreases gradually, and when the embedding size increases to 32, the LogLoss reaches the minimum value. Whether it is LogLoss or the AUC value, when the embedding size is too large or too small, the prediction accuracy will decrease. When the embedding size is too large, the transformation characteristics are too complex, which increases the difficulty of model training and affects the prediction effect. When the embedding size is too small, the compressed vector will lose a lot of original information.

The number of neural network layers greatly affects the performance of the model in deep learning. In this experiment, different network layers are set to verify its influence on the model performance. In order to eliminate the chance of a single algorithm, this paper does the same experiment on the NFM algorithm. The result is shown in Figure 11.

From Figure 11, under the condition of the same network depth, CAN and NFM show great differences in different test scenarios. However, the LogLoss values of the two experiments are also quite different, which shows that there are differences in the stability of the algorithm. At all network depths, the LogLoss and AUC of CAN are better than NFM, which proves the high efficiency of CAN again. Besides the number of neural network layers, the number of neurons in each layer also affects the efficiency of the algorithm. In this paper, the influence on the prediction effect is explored by changing the number of neurons in each layer. The number of neural networks is set to 2 layers, and the results are shown in Figure 12.

It can be concluded from the graph that the number of neurons has relatively little influence on the recommendation accuracy, and the index changes little under the same scale. The AUC value of CAN is slightly higher than that of NFM, but the LogLoss and the AUC index of the two algorithms are basically the same from the macroscopic point of view, which shows that increasing the number of neurons cannot increase the performance of the network. From Figure 12, the instability of CAN and NFM is also different under the condition of the same number of neurons. In one test, the overall performance of CAN is better than that of NFM, while in the other experiment, the performance of NFM is better than that of CAN in 150–500 neurons.

Figure 10: Effect of the embedding size on the coupling model of the advertising management module.

Figure 11: Influence of network depth on the effect of the coupling model of the advertising management module.
5. Conclusion

With the rapid development of the Internet, the research and application of the coupling relationship model of advertising management modules will become a hot spot. In practical application, Internet advertising has the characteristics of high dimension, sparse distribution, and fast update. According to the requirements of the product update rate and real-time, the advertising management module is divided into the offline recommendation algorithm based on the building training model and the online recommendation algorithm based on decision-making. The research work of this paper is mainly based on the prediction of the click-through rate of advertisements. By analyzing and mining the characteristics of users and products in real datasets, the datasets are preprocessed. Based on the attention mechanism and the deep neural network, this paper proposes an online advertising recommendation algorithm based on Gaussian process. Based on the characteristics of users and products, the advertising management module coupling based on the machine learning model provides users with the most reasonable advertising recommendation.

Data Availability

The experimental data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding this work.

Acknowledgments

This work was supported by the Special Research and Development and Promotion Project of Henan Province under Grant no. 212102210492 and the Science and Technology Planning Project of Nanyang under Grant no. KJGG102.

References


