Click-Through-Rate Prediction Using Deep Neural Networks and Efficient Channel Attention Mechanisms

Li Li
Anyang Normal University, Anyang 455000, China
E-mail: vvvlily@163.com

Keywords: deep neural networks, click-through prediction, online advertising, attention module, feature interaction

Received: Januar 2, 2025

The progress of technology and the Internet has brought the development of online advertising into a developed period. To optimize the accuracy of advertisement, click-through-rate rate, the research proposed an advertisement click-through-rate prediction model based on deep neural network combined with efficient channel attention network. This model consists of four parts: embedding layer, interaction layer, efficient channel attention layer, and prediction layer. The embedding layer is responsible for passing the feature vectors to the interaction layer. higher-order feature interactions are learned through deep neural networks and efficient channel attention networks are introduced for lower-order feature interactions. higher-order feature interactions can capture the nonlinear and complex relationships between original features, while low-level feature interactions mainly focus on the relationships between a few features. The channel attention layer integrates the original features with the features that have already been interacted with by the interaction layer. The prediction layer uses perceptrons to predict click-through-rates. The proposed model is compared with logistic regression, deep feature crossover network, and deep factorization machine on Criteo, Avazu, KDD12, and MovieLens-1M datasets. The results showed that when the network depth was 1, the area under the curve of the proposed model was 0.8377, which was 10.4% higher than that of the logistic regression model. The average logarithmic loss was 0.1985, which was lower than that of the comparison model. The UC value of the model in the KDD12 dataset was 0.7879 and the logarithmic loss value was 0.4478. Taken together, the proposed model of the study is able to predict click-through-rates more accurately and has better model performance.

Povzetek: Članek predstavi model za napovedovanje stopnje klikov (CTR) v spletnih oglasih, ki združuje globoke nevronske mreže in učinkovite mehanizme pozornosti, kar izboljša napovedi.

1 Introduction

The Internet has made an increasing amount of people tend to use the Internet to solve the items and other needs needed in daily life [1]. Today's Internet has penetrated into all aspects of life in various families, text message exchanges between people, online shopping, cell phone taxi travel and so on. Among the various types of service companies on the Internet revenue is mainly dependent on advertising revenue. Internet online advertising can rely on the huge user traffic, which can be transformed into operating income [2]. With the development of the Internet, the media used for advertisements have gradually evolved from television, magazines, and so on to the Internet, and online advertisements are commercial advertisements that are published in various websites with pictures, text, and so on [3]. For online advertising, how to send the advertisements to the people who need them is the main problem, and thus the click-through-rate (CTR) prediction, which is the most common way to calculate the effectiveness of the advertisement delivery, has been developed [4]. The Internet has made an increasing amount of people tend to use the Internet to solve the items and other needs needed in daily life [1]. Today's Internet has penetrated into all aspects of life in various families, text message exchanges between people, online shopping, cell phone taxi travel and so on. Among the various types of service companies on the Internet revenue is mainly dependent on advertising revenue. Internet online advertising can rely on the huge user traffic, which can be transformed into operating income [2]. With the development of the Internet, the media used for advertisements have gradually evolved from television, magazines, and so on to the Internet, and online advertisements are commercial advertisements that are published in various websites with pictures, text, and so on [3]. For online advertising, how to send the advertisements to the people who need them is the main problem, and thus the click-through-rate (CTR) prediction, which is the most common way to calculate the effectiveness of the advertisement delivery, has been developed [4]. For Internet service companies, CTR prediction is very important. In practical applications, CTR prediction models usually need to deal with the characteristics of high sparsity, which challenges the training and generalization ability of models, leading to the unsatisfactory prediction performance of models [5]. By predicting and analyzing the CTR of the advertisements, it can improve the accuracy of the advertisements that Internet service companies push to the people in the network, and it can also predict in advance the return rate that the advertisements pushed can bring back [6]. However, traditional CTR prediction models have limitations in handling large-scale sparse data and capturing complex feature interactions, which cannot fully capture nonlinear relationships and higher-order feature (HOF) interactions in the data, resulting in poor generalization ability and low prediction efficiency of the model. In this context, the study aims to address the low accuracy and efficiency of traditional CTR prediction methods by proposing the use of deep neural network (DNN) combined with efficient channel attention network (ECANet) for CTR prediction. The research aims to improve the accuracy of CTR prediction by proposing a new deep learning model, which can help advertisers plan and optimize their advertising strategies more effectively, and contribute to the long-term stable development of e-commerce.

There are two main innovations of this study. The first point is to combine DNN and ECANet for feature interaction, which enables the model to comprehensively understand and utilize feature information, thereby improving the accuracy of CTR prediction. The second point is to effectively fuse the original features with the interactive features using channel attention layers, and adaptively adjust the Fws through attention mechanisms, further improving the predictive performance of the model.

The main contribution of this study is: (1) proposing a deep learning model architecture that integrates DNN and ECANet, which helps improve the accuracy of advertising CTR prediction. (2) Introducing an ECANet and enhancing the expressive ability of key feature channels through dynamic weight allocation solves the problem of traditional models lacking adaptive selection for low-level feature interactions, while improving the interpretability of the model.

2 Related work

As the Internet has grown, so too has online advertising, and predicting the CTR of advertisements has become increasingly crucial. Advertising CTR prediction is one of the core algorithms in advertising technology, mainly used to predict the probability of ads being clicked. Traditional CTR prediction methods include logistic regression (LR), deep factorization machine (DeepFM), and deep feature crossover network (DCN). Jia S stated that CTR is closely related to students' level of active learning, reflecting their learning behavior in the data. Therefore, for the analysis of student performance data in online teaching, LR algorithm was proposed to predict whether students will pass the exam. The experimental results proved the feasibility of the proposed method [7]. Li et al. proposed a CTR prediction model based on DeepFM and used focal loss as the loss function. The results showed that the area under the curve (AUC) of the proposed model was 0.044 and 0.013 higher than that of the logistic model and neural network, respectively [8]. Huang G et al. improved the predictive performance of CTR based on DNN by introducing a regularized leader following the DCN model. The results indicated that the proposed model had good predictive performance and helped promote the development of e-commerce [9]. However, traditional LR, DeepFM, and DCN also suffer from poor generalization ability and low efficiency, making it difficult to meet the high efficiency and precision requirements in network environments. With the rapid development of computer technology, more and more intelligent algorithms are being applied to the field of CTR prediction. Mao K et al. proposed a dual-stream feature interaction model based on multi-layer perceptron (MLP) for CTR prediction in online advertising and recommendation, which enhances CTR modeling through feature selection module and group bilinear fusion module. The results indicated that the proposed model achieved competitive performance compared to many existing dual-stream CTR models [10]. Qin J et al. proposed a user behavior retrieval framework that supports arbitrary and learnable retrieval functions to address the long-term dependency issue of current CTR prediction models. The results indicated that the proposed method achieved a 6.6% Effective Cost Per Mille gain in A/B testing [11]. Li X et al. proposed a decision context interaction network to learn decision context in response to the problem that existing CTR prediction methods often ignore the information background that affects users' click decisions. The results indicated that the proposed model could effectively improve the accuracy of CTR prediction [12]. The summary table of the relevant research on CTR prediction methods mentioned above is shown in Table 1.

Table 1: Summary table of related research

Literature	Method	Advantage	Limitation
[7]	LR	Simple and computationally efficient	Unable to capture complex interactions between features
[8]	DeepFM	AUC is 0.044 and 0.013 higher than the logistic model and neural network model, respectively	Poor performance when dealing with very sparse data
[9]	DCN	Enhancing the learning ability of feature interaction by introducing cross networks	There are limitations in terms of representativeness

[10]	Feature selection module and group bilinear fusion module	Capable of effectively handling feature interactions	Mainly relying on feature selection and bilinear fusion, there are limitations in handling high-order feature interactions
[11]	Learnable retrieval function	Can effectively process user behavior data and improve the long-term dependence of CTR prediction	The main focus is on user behavior retrieval, therefore there are limitations in handling low-level feature interactions
[12]	Decision context interaction network	Can effectively improve CTR prediction accuracy	Mainly focusing on the interaction of decision context, there are limitations in handling high-order feature interactions

ECANet is an efficient channel attention module that adopts a local cross channel interaction strategy without dimensionality reduction, avoiding information loss caused by dimensionality reduction. It is commonly used to improve the performance of deep learning models. Gao Q et al. proposed to merge the Neck part of ECANet and YOLO v5 network to build an improved YOLO v5 network-based object recognition model for pedestrian target tracking and recognition in complex scenes. The results showed that the average accuracy of the proposed model improved by 1.3% [13]. Guo Z et al. proposed a lightweight semantic segmentation algorithm based on multi-module fusion, which combines coordinate attention and ECANet to enhance salient features, to address the problem of unsatisfactory and inefficient semantic segmentation results in existing methods. The results showed that the average pixel accuracy of the proposed algorithm reached 85.23%, and the training speed was improved by 68.69% [14]. Cui Z et al. proposed an image deblurring aggregation network based on ECANet to address the problem of low feature extraction efficiency in existing multi-scale network image denoising methods. The network improved feature representation through three feature aggregation blocks. The results indicated that the proposed method could clearly restore the texture and color of the image [15]. Chen W et al. proposed a low contrast defect detection method based on deep learning for ceramic parts, and used ECANet to detect defects in curved parts. The results indicated that the prediction accuracy of the proposed method could reach 94.35%, with an average detection time of only 0.78 seconds [16].

From domestic and international research, the traditional deep learning network for the prediction of advertisement CTR has the problems of low prediction accuracy, small scope of application, and inability to make long-term prediction. The study proposed a method based on DNN combined with ECANet for advertisement CTR prediction, which is expected to enhance the accuracy of advertisement CTR prediction and achieve long-term prediction.

3 CTR prediction using DNN combined with ECANet

3.1 DNN based CTR prediction

CTR prediction is the process of predicting the CTR of the next advertisement that has a probability of being clicked by a web user under a specific conditional situation using an algorithm before clicking on the advertisement [17]. The advertisement CTR prediction process is in Figure 1.

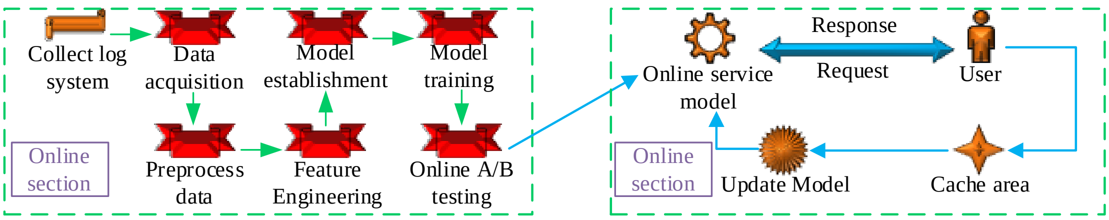


Figure 1: CTR predication process.

In Figure 1, the offline system collects and processes the data and trains the data, and passes the trained data into the online system, which delivers the advertisements to the user side through data caching. In traditional CTR prediction, the main role of feature engineering is to improve the prediction accuracy, which is improved by combining features to mine the potential relationship between different features. The CTR is calculated as Eq. (1).

$$CTR = \frac{num\_click}{num\_impression} \quad (1)$$

In Eq. (1) num_click means the quantity of clicks made by the web user and $num\_impression$ is the number of times the advertisement appears on the web user interface.

However, in actual user usage, it is possible that the web user does not make a click and thus there is a smoothing way to perform the formula leveling as in Eq. (2).

$$CTR = \frac{num\_click + \alpha \times \beta}{num\_impression + \beta} \quad (2)$$

In Eq. (2), α and β denote the smoothing parameters to avoid the problem that the denominator is zero leading to the inability to calculate the CTR, which can help the model to give reasonable predictions even in the case of sparse data. DNN is used to learn higher order feature interactions. In deep structural models, the DNN structure is divided into multi-layer networks, able to construct non-linear higher order features. This feature also makes DNN has a better performance in click rate prediction. DNN is more common in deep learning. In this problem of click prediction, the perceptron with additional layers in DNN is defined in equation (3).

$$f(x) = sign(\sum_{i=1}^{n} w_i x_i + b) \quad (3)$$

In Eq. (3), $sign$ denotes the sign function, n denotes the quantity of perceptron layers, x_i denotes the i-th input feature, w_i denotes the weight of x_i, and b denotes bias. Among them, input features refer to the raw data points used for training and predicting models. In the scenario of advertising CTR prediction, input features can include user features, advertising features, contextual features, historical behavior features, and interaction features. Weight is a parameter in DNNs that connects the input and output layers (OLs), representing the degree of influence of corresponding input features on the model output. During the training, weights are continuously adjusted through optimization algorithms to minimize the difference between predicted and actual values. This study uses the Adam optimization algorithm. In DNN, the output of the perceptron can be represented as a binary classification result, and the sign function is used to convert the output of the perceptron into a binary output sign function. The sign function is expressed as in Eq. (4).

$$sign(z) = \begin{cases} -1, x < 0 \\ 1, x \geq 0 \end{cases} \quad (4)$$

In Eq. (4), z denotes the input features. DNN increases the network layers on the perceptual machine, and to allow the model to learn HOFs, a hidden layer (HL) is added between the original input and OLs on the network model [18]. It makes the DNN can be diverse in choosing activation functions, and can choose more applicable activation functions according to the scene. To make the DNN capable of handling both regression and classification problems, neurons are added to the OL so that the OL of the DNN is no longer a single neuron [19]. Its DNN model network structure as in Figure 2.

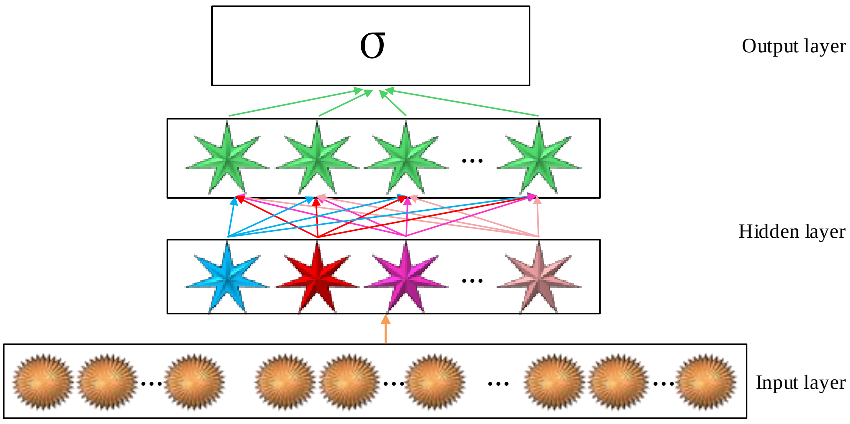


Figure 2: DNN model network structure.

In Figure 2, DNN has 3 layers: input, hidden, and output. The input layer (IL) sends the information feature collection to the HL, and after the training, the OL will work with the trained information features as output. The larger and more complex the dataset, the more HLs and units may be needed to capture patterns in the data. But more HLs and units also mean higher computational costs, and too many HLs and units may also lead to overfitting. Therefore, determining the optimal number of HLs and units usually requires experimentation and tuning. Assuming the number of HLs is L, the calculation of the input layer for the DNN model is shown in equation (5).

$$X = \{X_1, X_2, X_3, \ldots, X_n\} \quad (5)$$

In Eq. (5), X denotes the input vector. The HL is expressed as in Eq. (6).

$$h^{(l)} = h(W^{(l)}h^{(l-1)} + b^{(l)})(\forall l \in 1,2,3,\ldots,L-1) \quad (6)$$

In Eq. (6), $h^{(l)}$ and $b^{(l)}$ denote the input vector and bias vector of layer $l+1$ and l. $W^{(l)}$ denotes the weight matrix of layer $l-1$. The OL is represented as in Eq. (7).

$$y_{pre} = \arg\max_{C} P(y = C, X; w, b) \quad (7)$$

In Eq. (7), $P(y = C, X; w, b)$ denotes the possibility that the output category is equivalent to C, and the outcome category is either 0 or 1. In the forward propagation process of DNN, input features are input from the input layer, pass through multiple HLs, and finally output the predicted values of the model. It is necessary to optimize the objective function, and the minimum mean square error (MSE) is usually used to calculate the error between the predicted value and the true value, as shown in equation (8).

$$MSE = \frac{1}{M}\sum_{m=1}^{M}(y_m - \hat{y}_m)^2 \quad (8)$$

In Eq. (8), M denotes the number of samples. The activation value of the last HL a^H is a dense vector that contains all the information from the input layer to the HL. The study uses the Sigmoid function as the activation function for the OL, and introducing a^H into the Sigmoid function can generate the final predicted value, as shown in equation (9).

$$y_{DNN} = \sigma(W^{|H|+1} \cdot a^H + b^{|H|+1}) \quad (9)$$

In Eq. (9), $|H|$ is the number of layers of the HLs in the DNN. In the process of regression prediction, the loss function is expressed as in Eq. (10).

$$loss = \frac{1}{N}\sum_{i=1}^{N}(V_i - M_i)^2 \quad (10)$$

In Eq. (10), V_i denotes the predicted output and M_i denotes the actual output. In order for the loss function to become minimized, the network is to be trained and iteratively updated by Adam, expressed as in Eq. (11).

$$\begin{cases} v = \beta_1 v + (1-\beta_1)dw \\ s = \beta_2 s + (1-\beta_2)dw^2 \end{cases} \quad (11)$$

In Eq. (11), $\beta_1 = 0.9$, $\beta_2 = 0.999$. w denotes the changeover value and v, s denote the variables in the update process.

3.2 CTR prediction based on DNN-ECANet

Lower-order features (LOF) refer to basic features directly extracted from raw data, usually involving a single feature or a simple combination of a few features. They are rich in detail information and can capture subtle changes and local features in the data. HOF refer to features obtained through multiple nonlinear transformations and abstract representations of raw data, which can capture complex relationships and high-level semantic information in the data, usually involving interactions and combinations between multiple features. The study presents the ECANet in conjunction with DNN for the learning fusion of HOF and LOF interactions for advertisement CTR prediction. DNN is primarily used to perform HOF interactions, and its learning ability for LOF interactions is weak. DNN can automatically learn HOF interactions in data through multi-layer nonlinear transformations. In advertising CTR prediction, DNN can capture the complex relationships between user behavior, advertising attributes, and other features, providing richer information for prediction. The core of ECANet is the channel attention mechanism, which captures the dependency relationships between channels through one-dimensional convolution operations, enhancing the model's attention to important features. The combination of ECANet and DNN achieves dynamic modeling of multi-level feature interaction through efficient channel attention mechanism and complementary deep nonlinear learning. Compared with traditional attention mechanisms, ECANet proposes an adaptive method for selecting the size of one-dimensional convolution kernels, which avoids complex dimensionality reduction and enhancement processes and has higher computational efficiency. The ECANet is taken to learn the LOF interactions, and the problem of prediction for one in different application scenarios is different, for a specific CTR prediction, the ECANet is adopted to increase the feature weighting (Fw) and decrease the interference weights [20]. ECANet consists of three parts, as in Figure 3.

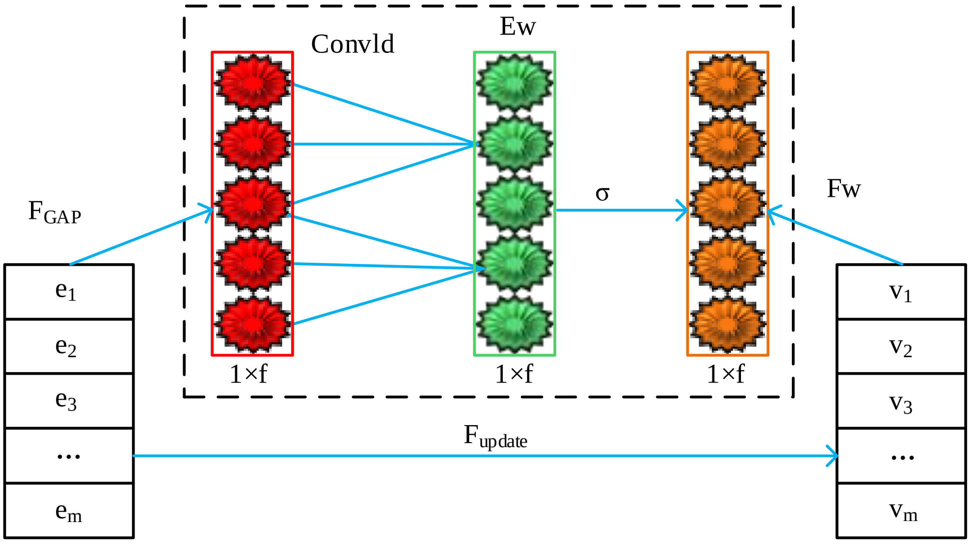


Figure 3: Composition of ECANet.

In Figure 3, ECANet consists of three parts: feature compression, weighting, and Weight extraction (Ew). ECANet takes the feature information and passes it to different network layers for weighting calculation by compressing the features. Finally, the weights are extracted by the weighting layer after the weighting calculation. Feature compression is used to get the received vector information and embed the vectors into ECANet [21]. It is calculated as in Eq. (12).

$$z_i = F(e_i) = \frac{1}{k}\sum_{t=1}^{k} e_i^{(t)} \qquad (12)$$

In Eq. (12), z_i denotes the statistical vector of the i -th input vector, $e_i^{(t)}$ denotes the t -dimensional information of i -th vectors, and $t = 1, 2, 3, \cdots, k$. To better capture the relationship between different features and improve the accuracy of advertisement CTR prediction, DNNs are utilized for advertisement CTR prediction, and the structure of DNN for CTR prediction is in Figure 4.

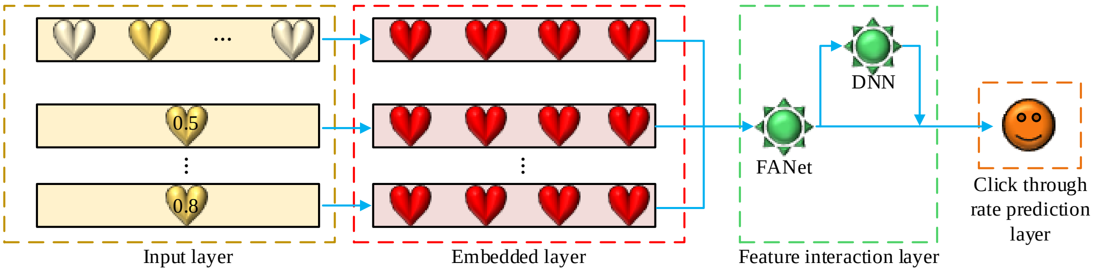


Figure 4: Structure of DNN for CTR prediction.

In Figure 4, DNN for click prediction has four parts, which are IL, embedding layer, feature interaction layer, and click prediction layer. The feature information is passed from the IL to the embedding layer, and the embedding layer further sends the processed features to the feature interaction layer through computation. The feature interaction layer integrates and interacts the received feature information, and finally the prediction layer makes the prediction. Ew is used to learn the weights of the embedding. The information is obtained using a single feature with the feature it wants to be close to, which is realized by convolution, and the weights of its field embeddings are calculated as in Eq. (13).

$$A = \sigma(conv1D_K(Z)) \qquad (13)$$

In Eq. (13), $conv1D$ denotes convolution, $A \in R^m$ denotes a vector, K denotes the size of the convolution kernel, and σ is the activation function. Fw is the most initial vector multiplied by the weights obtained from Ew, and thus the embedding vector obtained by fusion of the two, computed as in Eq. (14).

$$V = F(A, E) = [a_1 \cdot e_1, \cdots, a_m \cdot e_m] = [v_1, \cdots, v_m] \qquad (14)$$

In Eq. (14), $a_i \in R$, $e_i \in R^k$, $v_i \in R^k$. i denotes the i feature domain and k denotes the dimension of the embedding. In the established model, the DNN is interacted with the hidden features, which is calculated as in Eq. (15).

$$\begin{cases} x^1 = \sigma(W^{(1)}v + b^1) \\ x^n = \sigma(W^{(n)}v^{(n-1)} + b^n) \end{cases} \quad (15)$$

In Eq. (15), n denotes the neural network depth, x^n denotes the DNN output at layer n, and W denotes weight, and b denotes bias. In explicit feature interaction, ECANet is utilized. According to the DNN combined with ECANet for constructing the CTR prediction model, the study proposes that the structural composition of the DNN-ECA model is in Figure 5.

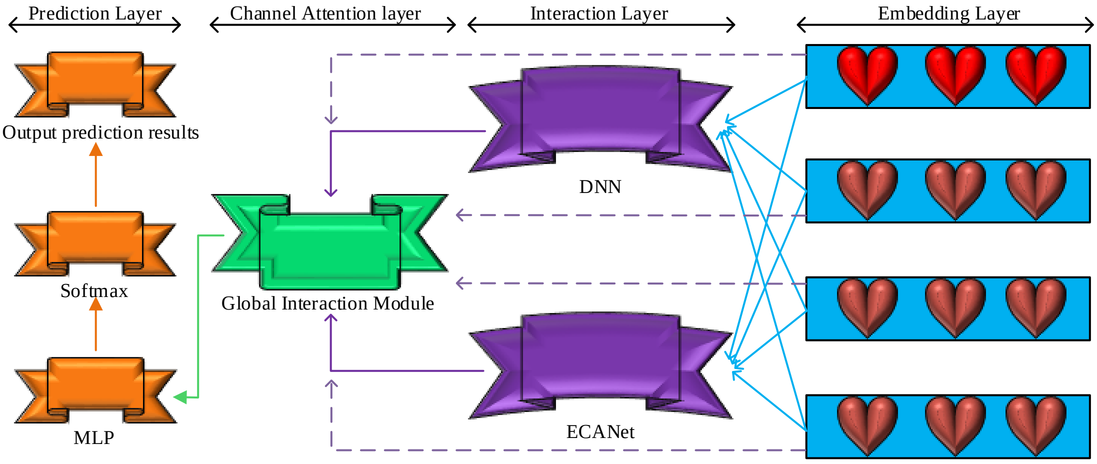


Figure 5: DNN-ECA model structure.

In Figure 5, four parts, the embedding layer, the interaction layer, the ECA layer, and the prediction layer, together form the DNN-ECA CTR model. Among them, the embedding layer maps the original feature vectors to a low dimensional embedding space for subsequent processing. The interaction layer consists of DNN and ECANet. DNN can capture complex feature combinations and learn high-order interaction relationships between features through multi-layer nonlinear transformations. ECANet focuses on learning lower-order interaction relationships between features. The efficient channel attention layer integrates the features output by the embedding layer with the features processed by the interaction layer, and adaptively adjusts the weights of different features through the attention mechanism to capture the relationship between the original features and the interaction features. The prediction layer consists of an MLP and a Softmax layer. MLP can perform deeper feature interaction and learning, while Softmax converts the output of MLP into a probability distribution for predicting CTR, ultimately outputting the predicted results for advertising CTR. That is, all the captured features are represented by a vector, as in Eq. (16).

$$r_g = [e_i; e_{ECA}; \tilde{e}_j] = [r_1; r_2; r_3] \quad (16)$$

In Eq. (16), r_1, r_2, and r_3 denote individual features. All the features are combined as inputs and the model output is calculated as in Eq. (17).

$$R_g = \sum_{l=1}^{L} \frac{exp(\tanh(r_l \cdot W_l + \hat{b}_l))}{\sum_{l=1}^{L} exp(\tanh(r_l \cdot W_l + \hat{b}_l))} r_l \quad (17)$$

In Eq. (17), W_l denotes the weights, b_l denotes the deviation matrix, and $\tanh$ denotes the activation function. In the final prediction layer, the adapted global features are delivered to the prediction module for CTR prediction using the perceptron machine, and the prediction process is in Eq. (18).

$$R_l = Relu(W_l R_l + b_l) \quad (18)$$

In Eq. (18), the value of l ranges from $l = 1, 2, \cdots, n$. For the lowest implicit layer, the final CTR prediction is performed for it, expressed as in Eq. (19).

$$\hat{y} = softmax(W_P R_h + b_q) \quad (19)$$

In Eq. (19), $\hat{y}$ denotes the prediction result, R_h denotes the output at layer h, which is the output of the prediction module, W_P represents the weight, and b_q denotes the deviation. In summary, the specific flowchart of the DNN ECANet model proposed by the research is shown in Figure 6.

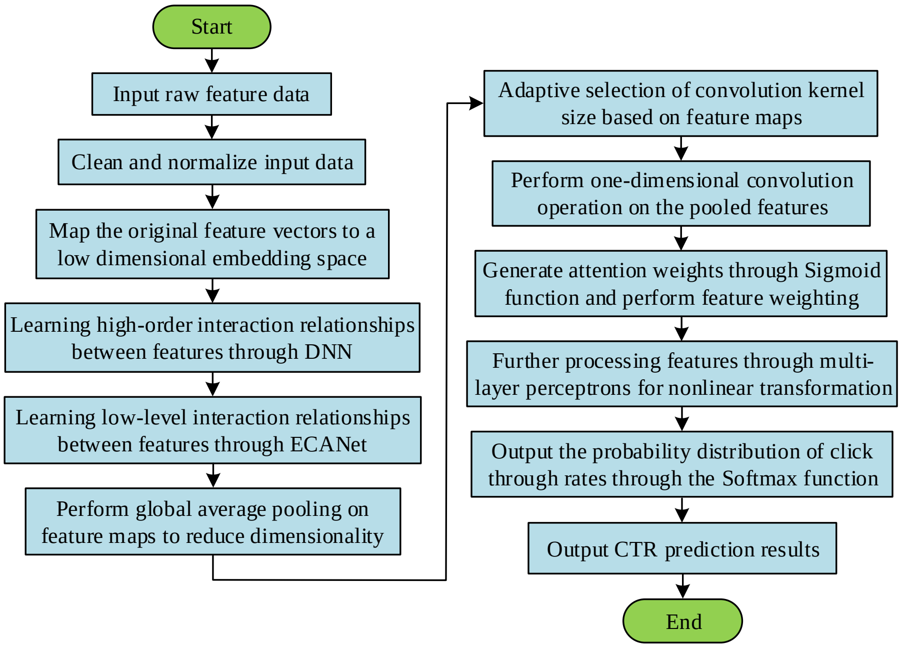


Figure 6: Flow chart of DNN ECANet model.

The study used AUC value and LogLoss as evaluation indicators. AUC is the area under the receiver operating characteristic (ROC) curve of the subject, and its value is positively correlated with the model performance. The horizontal and vertical axes of the ROC curve represent false accuracy and true accuracy, respectively. LogLoss is used to measure the distance between two components, and its calculation is shown in Eq. (20).

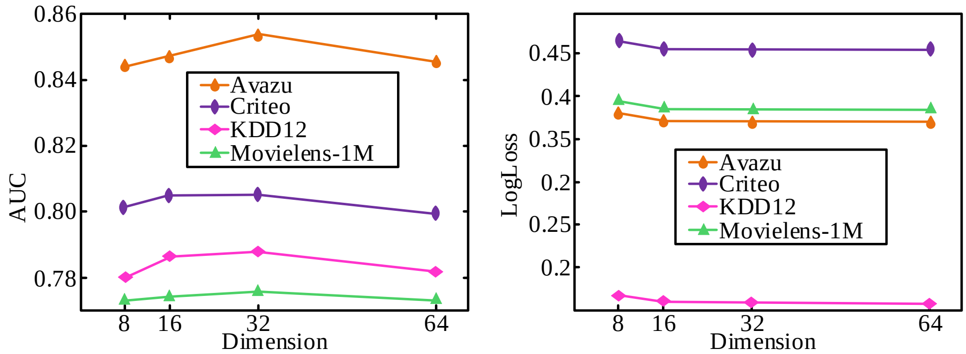
$$LogLoss = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)] \quad (20)$$

In Eq. (20), N denotes the total number of samples, y_i denotes the true value, and $\hat{y}_i$ denotes the predicted value.

4 Analysis of experimental results

4.1 Validity analysis of the DNN-ECANet Model

The experiment was run on the Windows 11 system and implemented in the TensorFlow 2.0 environment, AMD Ryzen 75800H CPU. The host memory was 32.0GB, and the graphics card was NVIDIA GeForce RTX 3060. The study used Python programming language, Softmax activation function, set layer depth to 3, number of neurons to 128, embedding dimension to 16, Drop Out Ratio to 0.7, learning rate to 0.001, batch size to 1024, and optimized the model using Adam. To analyze the predictive effectiveness of the DNN ECANet model, AUC and LogLoss were used as evaluation metrics in the study. AUC measured the probability that the predicted value of positive samples was higher than that of negative samples, and the larger the value, the better the predictive performance of the model. Logloss was the most important classification metric based on probability, used to measure the difference between predicted and true values. The smaller the value, the better the predictive performance of the model. The four datasets were Criteo, Avazu, KDD12, and MovieLens-1M. The Criteo dataset contains various information such as users, merchants, and advertisements, and is a classic dataset on consumer behavior and machine learning algorithms. The Avazu dataset provides rich information through multiple feature fields containing user characteristics and advertising attributes, and has classic application scenarios in the field of mobile advertising CTR prediction. The KDD12 dataset contains a large amount of user behavior data, which can help researchers evaluate and develop CTR prediction algorithms. The MovieLens-1M dataset contains user ratings, tags, and background information for movies. These four public datasets cover common application areas in different real-world scenarios, including data related to user preferences and behavior patterns, and can test the performance and generalization ability of the model under different conditions. Ten million data were randomly selected from each dataset and divided into training, testing, and validation sets in the ratio of 8:1:1. The experimental results are in Figure 7.



(a)The impact of embedding dimension on AUC

(b)The impact of embedding dimensions on LogLoss

Figure 7: Impact of embedding dimensions on the model.

From Figure 7 (a), as the embedding dimension increased from 8 to 32, the AUC values of the model gradually increased on all four datasets. When the embedding dimension reached 32, continuing to increase the embedding dimension would gradually decrease the AUC value of the model, and the performance would begin to decline. From Figure 7 (b), as the embedding dimension increased from 8 to 16, the LogLoss of the model gradually decreased, indicating that the prediction accuracy of the model gradually increased. When the embedding dimension reached 16, even if the embedding dimension continues to increase, there was no significant change in the LogLoss of the model. This may be because appropriately increasing the embedding dimension can enable the model to capture richer feature representations, thereby improving the model's ability to distinguish between positive and negative samples. But if the embedding dimension is too large, it may lead to overfitting of the model and a decrease in generalization ability on the test data. Therefore, it is reasonable to set the embedding dimension to 16 in the study. To further investigate the impact of the number of layers and neurons in each layer of DNN on the robustness of the model, the layers were set to 2, 3, and 4, and the number of neurons was set to 64, 128, and 256, respectively. The changes in AUC and LogLoss values of the DNN-ECA prediction model proposed by the research on the Criteo dataset are shown in Table 2.

From Table 2, as the number of layers increased, the AUC and LogLoss values of the model did not show a significant monotonic trend. This indicated that the performance of the model did not solely depend on an increase in the number of layers.

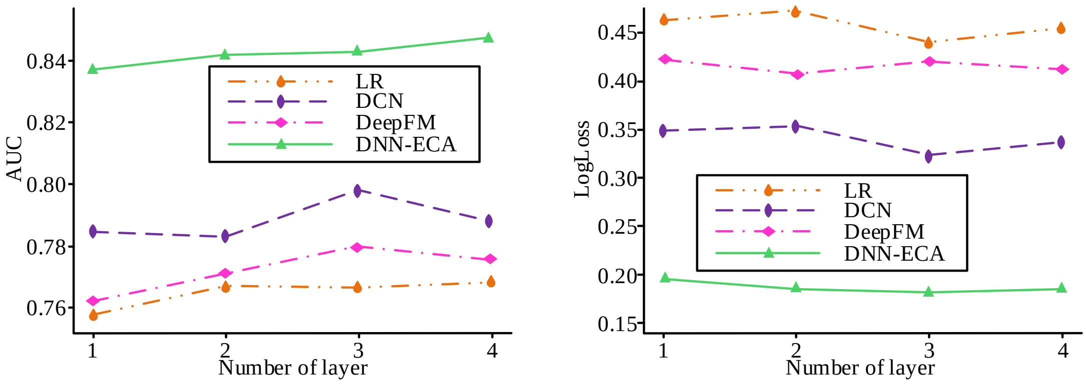
Table 2: AUC and LogLoss values of DNN-ECA model under different hyperparameters

Number of layers	Number of neurons	AUC	LogLoss
2	64	0.821	0.453
2	128	0.825	0.439
2	256	0.818	0.452
3	64	0.831	0.435
3	128	0.837	0.428
3	256	0.829	0.437
4	64	0.824	0.441
4	128	0.832	0.436
4	256	0.829	0.438

In addition, when the number of layers was fixed, increasing the number of neurons usually increased the AUC value of the model and reduced the LogLoss value.

When the number of layers was 3 and the number of neurons was 128, the model performed the best in terms of AUC and LogLoss values, which were 0.837 and 0.428, respectively.

Therefore, it is reasonable to set the layer depth to 3 in the study. The LR model, DCN model, and DeepFM are three classic CTR prediction models that are representative and highly applicable, providing comprehensive comparisons and references for this study. The performance of DNN-ECA was compared with LR model, DCN model, and DeepFM at different network depths, and the experimental results are in Figure 8.



(a)The impact of network depth on AUC (b)The impact of network depth on LogLoss

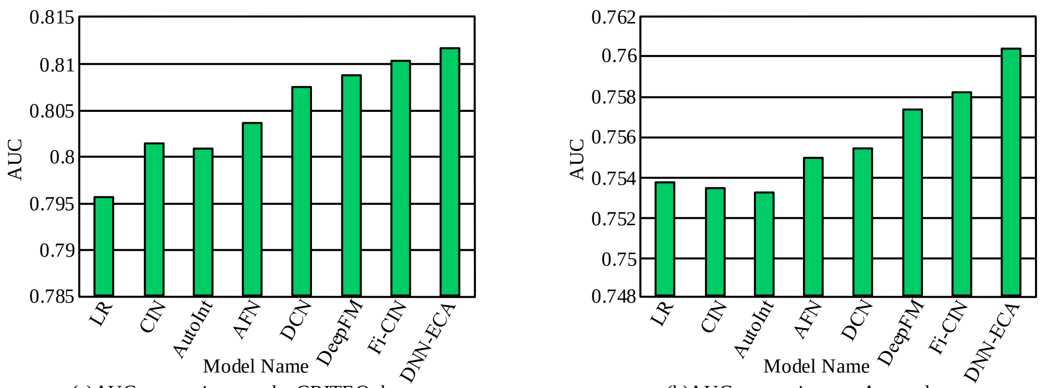
Figure 8: The influence of network depth on the model.

From Figure 8(a), as the network depth increased, the AUC value of the DNN-ECA model proposed by the research also increased continuously. When the network depth was 1, the AUC value of the proposed model was 0.8377, which was 10.4% higher than that of the LR model. The average AUC value of the proposed model was 0.845, which was higher than the comparison model. The average AUC value of the LR model was the lowest, at 0.765. From Figure 8 (b), at different network depths, the LogLos of the proposed model was always the lowest and the variation amplitude was small, with an average LogLos of 0.1985. The LogLos of the other three prediction models fluctuated continuously with the increase of network depth, which greatly affected the predictive performance of the models. The results indicated that the DNN-ECA prediction model proposed by the research had better predictive performance than LR, DCN, and DeepFM models, and was less affected by network depth. Further comparing the computational complexity of the four models mentioned above, the results are shown in Table 3.

Table 3: Comparison of computational complexity among four models

Model	Number of parameters/M	Training time/s
LR	0.01	150
DCN	1.22	163
DeepFM	1.54	201
DNN-ECA	1.36	179

From Table 3, the parameter of DNN-ECA was 1.36M and the training time was 179s, which was superior to the DeepFM model in terms of computational complexity. In the public dataset Criteo, the DNN-ECA was compared with the commonly used CTR prediction model for machine learning performance. Figure 9 shows the comparative results.



(a)AUC comparison on the CRITEO dataset (b)AUC comparison on Avazu dataset

Figure 9: Comparison of AUC performance of different models on the dataset.

Figure 9(a) shows that when comparing the model performance in the Criteo dataset, the DNN-ECA model had the highest AUC of 0.8124. While the commonly used LRM had an AUCof 0.7951, which was 0.0173 lower than that of the DNN-ECA. This indicated that compared to commonly used prediction models in the Criteo dataset, DNN-ECA had the best predictive performance. In Figure 9(b), the Avazu dataset, the DNN-ECA model had the highest AUC value of 0.7612, and the LRM had an AUC value of 0.7538, which was 9.8% lower than the AUC of the DNN-ECA. the AutoInt model had the lowest AUC value of 0.7531. This showed that the DNN-ECA had the best prediction performance in the dataset Avazu. Set Avazu had the best prediction performance, higher AUC value than the commonly used CTR prediction model, and better performance than the classical deep learning prediction model. In Criteo, Avazu and Movielens-1M datasets, the DNN-ECA and the commonly used CTR prediction model were subjected to log-loss experiments. Table 4 lists the specific data.

Table 4: Performance comparison of various models on the dataset

Model	Criteo	Avazu	Movielens-1M
	LogLoss	LogLoss	LogLoss
LR	0.4756	0.1679	0.4416
CIN	0.4695	0.1598	0.4269
AutoInt	0.4560	0.1581	0.4273
AFN	0.4471	0.1569	0.4149
DCN	0.4416	0.1589	0.4035
DeepFM	0.4511	0.1588	0.3891
Fi-CIN	0.4483	0.1544	0.3843
DNN-ECA	0.4462	0.1539	0.3795

From Table 4, the LogLoss of the proposed model on the Criteo dataset was 0.4462, which was 0.0294 lower than that of the LR model. The LogLoss on the Avazu dataset was 0.1539, which was 0.005 lower than the DCN model. The LogLoss on the Movielens-1M dataset was 0.3795, which was 0.024 lower than the DeepFM model. To further investigate the impact of the number of input fields on the performance of DNN-ECA, the Criteo dataset was used to compare DNN-ECA with DeepFM. The experimental results are shown in Figure 10.

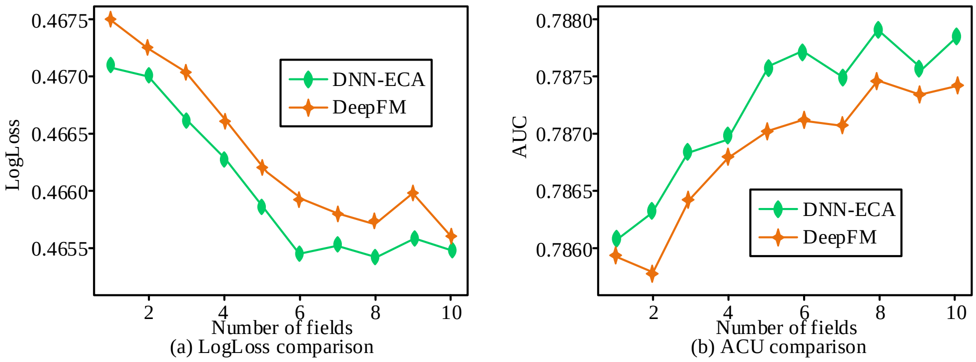


Figure 10: Influence of different contextual numbers on the model.

From Figure 10 (a), the LogLoss value of the proposed DNN-ECA model was consistently lower than that of the DeepFM model under different input field quantities. When the number of input fields was 8, the LogLoss of the model was the lowest, which was 0.4654. The highest LogLossof the DeepFM model was 0.4675, and the lowest LogLoss was 0.4656. From Figure 10 (b), the lowest AUC value of the DNN-ECA model was 0.7862, and the highest AUC value was 0.7879. Under different input field quantities, the AUC value of the DNN-ECA model was consistently higher than that of the DeepFM model. It demonstrates that the DNN-ECA has more learning ability and is better for the evaluation of prediction. To investigate the impact of ECANet on model performance, ECANet was removed from the DNN-ECA model and the AUC values of the complete model and the model were compared with ECANet removed in four datasets. The results are shown in Figure 11.

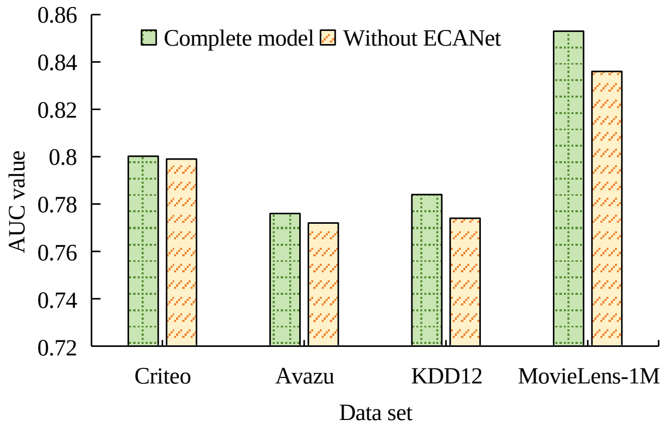


Figure 11: Impact of ECANet on model performance.

From Figure 11, in all four datasets, the AUC values of the complete model were higher than those of the model with ECANet removed, indicating that ECANet can effectively improve the CTR prediction performance of the model.

4.2 Validation of the effectiveness of DNN-ECA CRT Prediction

To verify the practical application effect of the proposed DNN-ECA model, data collection was conducted on the campus forum network of Sichuan Agricultural University. Information was collected from 200 users and 500 posts from October 1, 2024 to October 31, 2024. The collected data were stored in a MySQL database. Duplicate user and post information were removed through unique identifiers such as user ID and post ID. Operations such as word segmentation, stem extraction, and morphological restoration were performed on post and comment content to extract meaningful features. The DNN-ECA model was compared with four commonly used deep learning models: Factorization-machine supported Neural Networks (FNN), Neural Factorization Machines (NFM), Wide and Deep Learning Model (Wide&Deep), and DeepEM. The FNN model consisted of a factorization machine (FM) and an MLP, and has good feature cross expression ability. The NFM model integrated FM and DNN through a dual line interactive pooling layer. The Wide&Deep model combined LR and DNN for learning. The comparison results of the indicators of the five models are shown in Figure 12.

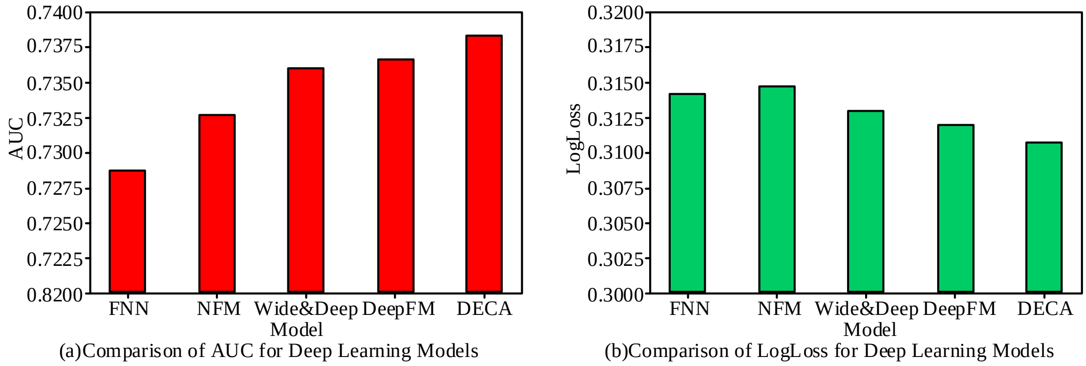


Figure 12: Comparison between DNN-ECA model and deep learning model.

From Figure 12(a), results indicated that the highest and lowest AUC value of the DNN-ECA and FNN were 0.7379 and 0.7286. For the traditional DeepFM for CTR prediction, the AUC value was 0.7361. It was 0.011 lower than that of the AUC of the DNN-ECA. The DNN-ECA had the best performance and higher accuracy of CTR prediction compared to the rest of the deep models. In Figure 12(b), the DNN-ECA model had the lowest LogLoss value of 0.3119, which was 0.349% lower than that of the DeepFM, and 0.547% lower than NFM model. It indicated that the proposed DECA performed better compared to other deep learning models commonly used for CTR prediction. The DNN-ECA model could simultaneously perform the fusion of learning different higher and lower-order features. The comparison of CTR prediction of different models on KDD12 dataset is in Figure 13.

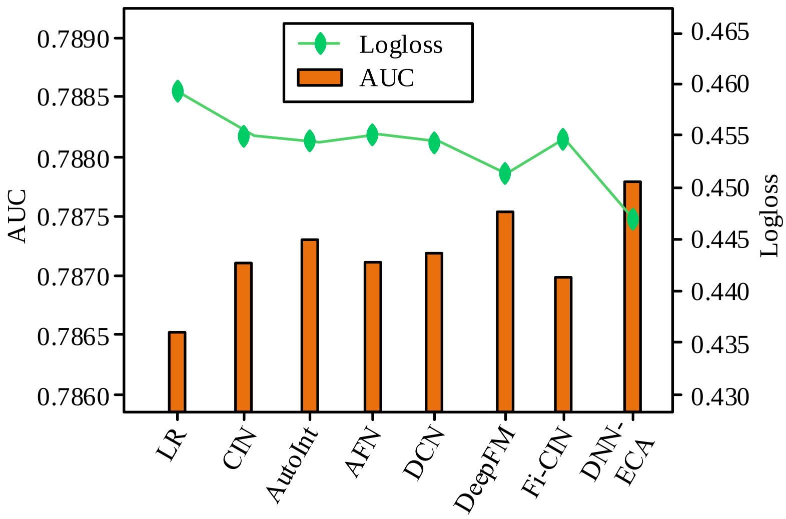


Figure 13: Comparison of CTR prediction of models on KDD12.

In Figure 13, when different models were used for CTR prediction on the KDD12 dataset, the LRM had the worst CTR prediction performance, with an AUC value of 0.7866 and a LogLoss value of 0.4587. The DeepFM had a better CTR prediction performance compared to the rest of the commonly used prediction models, with an AUC of 0.7876 and a LogLoss value of 0.4529. The DNN-ECA had the best CTR prediction performance with an AUC of 0.7879 and a LogLoss value of 0.4478. Compared to the LRM, the AUC value improved by 0.0013 and the LogLoss value reduced by 0.24%. Compared with the DeepFM, the AUC increased by 0.003 and the LogLoss value decreased by 0.19%. To further validate the superiority of the proposed model, the KDD12 dataset was used to compare the prediction accuracy of the DNN-ECA model with three advanced models: Automatic Feature Interaction (AutoInt), Feature Importance Bilinear Feature Interaction (FiBiNet), and Decision context interaction network (DCINet). The results are shown in Figure 14.

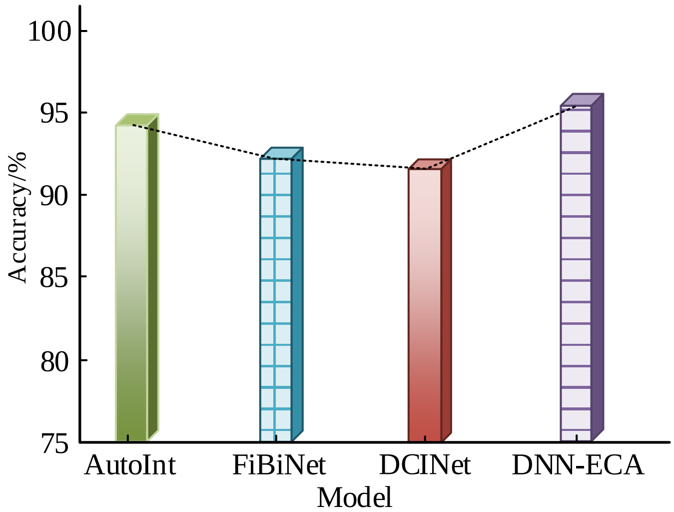


Figure 14: Comparison of prediction accuracy among four models.

From Figure 14, among the four models, the DNN-ECA model proposed by the research had the highest accuracy, at 96.89%. Next was the AutoInt model, while the DCINet model had the lowest accuracy, but still above 90%. The results indicated that the proposed DNN-ECA model had a high prediction accuracy and certain superiority.

5 Discussion

To optimize the accuracy of advertisement CTR prediction, the study proposed a method grounded on the fusion of DNN and ECANet for CTR prediction. The results demonstrated that on different datasets, when the embedding dimension of the model was 32, the AUC value of the model was the highest and the prediction performance was the best. Continuing to increase the embedding dimension, the AUC value of the model began to decrease. This meant that the model could achieve good performance at lower embedding dimensions. Continuing to increase embedding dimensions may cause the model to start memorizing specific samples in the training data instead of learning generalized feature representations, resulting in overfitting and high complexity, which can affect its generalization ability. In addition, the DNN-ECA model proposed by the research performed better on the Criteo dataset, possibly because the Criteo dataset typically contains a large number of sparse features. When the embedding dimension was 32, the model had sufficient feature representation ability to capture complex relationships between sparse features.

The DNN-ECA had a lower LogLoss compared to the LRM. In the Avazu dataset, the LogLoss of the DNN-ECA was 0.014 lower than the LRM and 0.005 lower than the DCN, and the AUC value was 0.017 higher than the LRM and 0.0061 higher than the DCN. Under different input field quantities, the LogLoss value of the DNN-ECA model was consistently lower than that of the DeepFM model. When the number of input fields was 8, the LogLoss of the model was the lowest, which was 0.4654. In the KDD12 dataset, the AUC value of the DNN-ECA model was 0.7879, and the LogLoss value was 0.4478. Compared to the LR model, the AUC value increased by 0.0013 and the LogLoss value decreased by 0.24%. DNN-ECA model proposed by the research had the highest accuracy, at 96.89%. Next was the AutoInt model, while the DCINet model had the lowest accuracy, but still above 90%. The DNN-ECA model proposed by the research exhibited excellent performance compared to traditional LR [7], DeepFM [8], and DCN [9], mainly due to the model's use of DNN, ECANet, and channel attention to achieve cross level feature interaction fusion. In addition, the attention mechanism adopted by the DNN-ECA model adaptively captured the dependencies between channels through one-dimensional convolution, without the need for dimensionality reduction or enhancement, preserving the information integrity of the original channel features and reducing model complexity and computational burden.

6 Conclusion

In summary, the study's proposed DNN-ECANet method for advertisement CTR prediction can improve the prediction accuracy and help the long-term development of the advertising industry in the Internet environment. The DNN-ECA model proposed in the study has great potential for application in niche markets and specific time environments, and is expected to provide more accurate advertising strategies for advertisers and platforms. Specifically, in niche markets, advertisers often face issues such as small data volumes and specific user groups. By optimizing feature engineering and introducing multimodal data, the DNN-ECA model can better adapt to the needs of niche markets and improve the accuracy of advertising placement. In addition, user behavior patterns may change during specific time periods, and the DNN-ECA model can optimize advertising placement in real-time and improve CTRs by introducing time series features and dynamically adjusting strategies. Although the DNN-ECA model proposed in the study can effectively capture higher-order and lower-order feature interactions, thereby improving CTR prediction performance, the complexity of the model is also high, resulting in longer training and inference times, and requiring a large amount of training data to achieve optimal performance. Therefore, in future research, methods for dynamically adjusting the learning rate, such as gradient descent-based dynamic adjustment strategy or learning rate annealing strategy, can be further explored to improve the convergence speed and stability of the model, or to further enhance the prediction performance by integrating multiple deep learning models.

References

[1] Yang Y, Xu B, Shen S, Shen F, Zhao J. Operation-aware Neural Networks for user response prediction. Neural Networks, 2020, 121(8): 161-168. https://doi.org/10.1016/j.neunet.2019.09.020

[2] Todri V, Ghose A, Singh P V. Trade-offs in online advertising: Advertising effectiveness and annoyance dynamics across the purchase funnel. Information Systems Research, 2020, 31(1): 102-125. https://doi.org/10.1287/isre.2019.0877

[3] Aribarg A, Schwartz E M. Native advertising in online news: Trade-offs among clicks, brand recognition, and website trustworthiness. Journal of Marketing Research, 2020, 57(1): 20-34. https://doi.org/10.1177/00222437198797

[4] Lyu Z, Dong Y, Huo C, Ren W. Deep match to rank model for personalized click-through rate prediction. In Proceedings of the AAAI Conference on Artificial Intelligence, 2020, 34(1): 156-163. https://doi.org/10.1609/aaai.v34i01.5346

[5] Yuan L, Pan Z, Sun P, Wei Y, Yu H. Deep context interaction network based on attention mechanism for click-through rate prediction. Journal of Intelligent & Fuzzy Systems, 2021, 41(6): 6899-6914. https://doi.org/10.3233/JIFS-210830

[6] Gai P J, Klesse A K. Making recommendations more effective through framings: Impacts of user-versus item-based framings on recommendation click-throughs. Journal of Marketing, 2019, 83(6): 61-75.

[7] Jia S. Logistic Regression Analysis of Online Course Click Rate. 2021 International Conference on Machine Learning and Intelligent Systems Engineering (MLISE). IEEE, 2021: 388-391. https://doi.org/10.1109/MLISE54096.2021.00081

[8] Li L, Hong J, Min S, Xue Y. A novel CTR prediction model based on DeepFM for taobao data. 2021 IEEE International Conference on Artificial Intelligence and Industrial Design (AIID). IEEE, 2021: 184-187. https://doi.org/10.1109/AIID51893.2021.9456556

[9] Huang G, Chen Q, Deng C. A new click-through rates prediction model based on Deep&Cross network.

Algorithms, 2020, 13(12): 342-342. https://doi.org/10.3390/a13120342

[10] Mao K, Zhu J, Su L, Cai G, Li Y, Dong Z. FinalMLP: an enhanced two-stream MLP model for CTR prediction. Proceedings of the AAAI conference on artificial intelligence. 2023, 37(4): 4552-4560. https://doi.org/10.1609/aaai.v37i4.25577

[11] Qin J, Zhang W, Su R, Liu Z, Liu W, Zhao G, et al. Learning to retrieve user behaviors for click-through rate estimation. ACM Transactions on Information Systems, 2023, 41(4): 1-31. https://doi.org/10.1145/3579354

[12] Li X, Chen S, Dong J, Zhang J, Wang Y, Wang X, Wang, D. Decision-making context interaction network for click-through rate prediction. Proceedings of the AAAI Conference on Artificial Intelligence. 2023, 37(4): 5195-5202. https://doi.org/10.1609/aaai.v37i4.25649

[13] Gao Q, He Z, Jia X, Xie Y. Han X. Lightweight high-precision pedestrian tracking algorithm in complex occlusion scenarios. KSII Transactions on Internet and Information Systems (TIIS), 2023, 17(3): 840-860. https://doi.org/10.3837/tiis.2023.03.009

[14] Guo Z, Ma D, Luo X. A lightweight semantic segmentation algorithm integrating CA and ECA-Net modules. Optoelectronics Letters, 2024, 20(9): 568-576. https://doi.org/10.1007/s11801-024-3241-z

[15] Cui Z, Wang N, Su Y, Zhang W, Lan Y, Li A. ECANet: Enhanced context aggregation network for single image dehazing. Signal, Image and Video Processing, 2023, 17(2): 471-479. https://doi.org/10.1007/s11760-022-02252-w

[16] Chen W, Zou B, Huang C, Yang J, Li L, Liu J, Wang X. The defect detection of 3D-printed ceramic curved surface parts with low contrast based on deep learning. Ceramics International, 2023, 49(2): 2881-2893. https://doi.org/10.1016/j.ceramint.2022.09.272

[17] Feng S, Zhou H, Dong H. Using deep neural network with small dataset to predict material defects. Materials & Design, 2019, 162(10): 300-310. https://doi.org/10.1016/j.matdes.2018.11.060

[18] Žvirblis T, Pikšrys A, Bzinkowski D, Rucki M, Kilikevičius A, Kurasova O. Data augmentation for classification of multi-domain tension signals. Informatica, 2024, 35(4): 883-908. https://doi.org/10.15388/24-INFOR578

[19] Mena-Yedra R, López Redondo J, Pérez-Sánchez H, Martinez Ortigosa P. ALMERIA: Boosting pairwise molecular contrasts with scalable methods. Informatica, 2024, 35(3): 617-648. https://doi.org/10.15388/24-INFOR558

[20] Hua J, Li X, Liu J, Tang J, Rao, J, Deng H. A novel arrhythmia classification of electrocardiogram signal based on modified HRNet and ECA. Measurement Science and Technology, 2022, 33(6): 701-713. https://doi.org/10.1088/1361-6501/ac51a3

[21] Duan X, Sun Y, Wang J. ECA-UNet for coronary artery segmentation and three-dimensional reconstruction. Signal, Image and Video Processing, 2023, 17(3): 783-789. https://doi.org/10.1007/s11760-022-02288-y