SELNN: Effective Crossover Features by Attention Mechanism and Logarithmic Conversion for CTR

Lu Meng¹, Li Wang*¹

¹ School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan, Liaoning, 114051, China
*Corresponding author's e-mail: wangli9966@ustl.edu.cn

Abstract. With the continuous increase of web resources, information overload is becoming more and more serious. Getting the information that meets the needs of a large amount of data has become an urgent problem. One of the popular studies in click-through rate prediction is to construct feature interactions to improve prediction accuracy. Traditional models with low-order interactions cannot adequately represent the intersection information, and deep models with undifferentiated feature intersections lack relevance. In this paper, we propose a model called SELNN, which improves the accuracy of click-through rate prediction through attention mechanism and logarithmic conversion. On the one hand, improving attention mechanisms learns the importance of each feature itself. On the other hand, combining logarithmic conversion structure with feedforward neural networks to learn different orders of feature interactions. The experimental results indicated that the AUC values of SELNN on the Criteo and Movielens-1M datasets were 81.46% and 87.36%, respectively. SELNN effectively improves the prediction accuracy and reduces the number of parameters and computational effort.

1. Introduction

Revenue from advertising has been growing in recent years. Estimating users' clicks on ads and then placing them precisely can greatly increase advertisers' revenue.

The prediction models currently proposed in academia and industry can be divided into two main categories. Traditional models as LR[1], FM[2] are simple and easy to reference. However, it is not possible to perform parallel training, and the cross-correlation between features is neglected. Deep learning-based models as PNN[3], AFM[4], DCN[5], DeepFM[6], and MiFiNN[7]. The Click-Through Rate is improved by strengthening the intensity of feature interactions, combining memory and generalization capabilities, introducing attention mechanisms. However, most models use undifferentiated feature interactions lacking relevance.

The SELNN proposed in this paper improves the accuracy of CTR prediction utilizing attention mechanism and logarithmic conversion. The main contributions of this paper are as follows:

- The SELNN cleverly applies the SENet from computer vision research to recommendation algorithms and improvements to achieve feature importance selection.
- The SELNN utilizes a logarithmic conversion structure to learn more patterns of feature interactions from valid feature combinations. It enables DNN to expand the input in potential mining information and then guide click-through prediction from feature interactions of different orders.
2. Related work

2.1. Click-through rate prediction
The user and item features are tandem concatenated as an input feature, denoted by $x \in \mathbb{R}^n$, where $n$ is the dimensionality of the tandem features. Traditional studies usually input $x$ directly into the model for prediction, such as LR\(^{[1]}\), FM\(^{[2]}\). However, because of the sparse and high-dimensional original vector $x$, the model is prone to overfitting. Therefore, the latest research focuses on extracting key information from feature interactions and making predictions.

2.2. K-order interactive feature
For the input feature $x \in \mathbb{R}^n$, the k-order feature interaction is defined as $g\left(x_{i_1}, \ldots, x_{i_k}\right)$. Where $g(\cdot)$ is a function of the feature interaction, which can be the inner product, outer product, and Hadamard product. Professionals usually perform traditional research for manual feature combinations, such as Wide&Deep\(^{[8]}\). However, it has poor generalization capability, requires more manual operations, and cannot be applied across domains. Therefore, the current popular research is to learn higher-order feature interactions using deep learning methods automatically.

2.3. Attention mechanism
The attention mechanism distributes computational resources to more important tasks by calculating attention weights. It can effectively solve the information overload problem. The features are filtered and judged according to the attention weights to determine the significant impact on the results. The current mainstream approach calculates the attention score based on the relevance of user attributes and item attributes for more targeted recommendations. Such as the application of self-attention\(^{[9]}\). However, its training process is complex with long latency and needs to be optimized.

3. Model description

![Figure 1 SELNN architecture.](image)
The SELNN architecture is shown in Figure 1. It consists of the components: input layer, SE embedding layer, compressed interaction network, logarithmic conversion layer, multiple hidden layers, and output layer. The data in the input layer has numerical features and categorical features. The SE embedding layer first converts the original sparse features into a vector of dense features, then the weight is learning for each feature by using the attention mechanism. This can suppress the effects of low frequency and invalid features. Next in parallel are the explicit feature interaction component and the implicit feature interaction component. The compressed interaction network implements the explicit feature interaction, combining logarithmic conversion structure with feedforward neural networks. Finally, the output layer predicts the click-through rate by the Sigmoid function. The following section describes the details of SELNN.

3.1. Input layer

The input data consists of sparse numerical and categorical features in the CTR prediction problem for real scenarios. Representing the classification features as one-hot vectors:

\[ x = [0, 1, 0, \ldots, 0], [1, 0], \ldots, [0, 1, 0, 1, \ldots, 0] \]  

where \( m \) is the total number of features and field \( i \) denotes the \( i \)-th feature.

3.2. SE embedding layer

The feature vectors are dynamically differentiated according to their importance, utilizing an attention mechanism. The learning procedure of SENet \([10]\) is divided into three main steps. First is Squeeze, which compresses the data and summarizes the information for each feature vector. Then comes Excitation, which introduces a narrow two-layer MLP network to learn a feature weight \( A = \{a_1, a_2, \ldots, a_i, \ldots, a_f\} \) for each feature, and finally is Re-Weight, which takes the values of \( f \) weights and multiplies them back into the corresponding feature vector. This completes the feature importance extraction.

3.3. Compressed interaction network

CIN follows the crossover network in xDeepFM \([11]\). It accomplishes higher-order explicit feature interactions by recursion. Its core is the summation and pooling of the individual hidden layer feature vectors:

\[ p_i^k = \sum_{j=1}^{H_k} x_{i,j}^k \]  

Where \( i \in [1, H_k] \). This gives the pooling vector for the \( k \)-th layer \( p_i^k = [p_1^k, p_2^k, \ldots, p_{H_k}^k] \). The pooling vectors of each layer are concatenated as \( p^+ = [p_1^1, p_2^1, \ldots, p_T^1] \in R^{\sum_{i=1}^{H} H_i}. \) From there, the output vector of the CIN component is obtained.

3.4. Logarithmic conversion network

The logarithmic transformation layer learns the power (order) of each feature in the feature interactions, and This layer is composed of multiple logarithmic neurons, each of which is calculated as

\[ y = \exp(\sum_{i} w_i \ln x_i) = \prod_i x_i^{w_i} \]  

The lack of MLP expressiveness is compensated by converting the input into logarithmic space. It should be noted that the logarithm takes values in the range \((0, +\infty)\). Therefore, a minimal positive number, \(1 e^{-\gamma}\) is added to the input to ensure that each input is positive.

Next, all the crossover features are connected, and multiple hidden layers are stacked and input into the feedforward neural network.

3.5. Output layer

The SELNN combined Linear, CIN, and LCN, then processed by Sigmoid function to get the final click prediction results. The calculation formula of SELNN is:
\[ \hat{y}_{SLENN} = \sigma(W_{\text{Linear}}^T \hat{y}_{\text{Linear}} + W_{\text{CIN}}^T p^+ + W_{\text{LCN}}^T \hat{y}_{\text{LCN}} + b) \] (4)

Where \( W_{\text{Linear}}^T \), \( W_{\text{CIN}}^T \) and \( W_{\text{LCN}}^T \) denote the weights of Linear, CIN, and LCN, respectively. \( b \) is the overall deviation of the model. The output of the Sigmoid function is between 0 and 1, which is convenient for the judgment of click-through rate prediction.

4. Experiments

4.1. Datasets
Criteo dataset contains records of 4500w user clicks on display ads over seven consecutive days, with 13 continuous numerical features and 26 discrete categorical features per sample. Movielens-1M database contains nearly 1 million rating data from 6040 users for 3706 movies.

4.2. Evaluation metrics
CTR prediction is a dichotomous problem. The evaluation metrics adapt AUC and Logloss. The higher the AUC value, the more accurate the prediction. The smaller the Logloss value, the better the performance of the model.

4.3. Hyper-parameter study

4.3.1. Embedding size. It is critical to the model's representation capability and computational cost. Therefore, Embedding sizes are set to 5, 10, 15, and 20; experiments are conducted separately to explore how their sizes affect the prediction performance. The results of the experiment are shown in Figure 2. The AUC and Logloss reach the optimal value on the two datasets when the embedding size is 15. Because it is too small, it will lose the crossover features, and too large will be overfitting.

4.3.2. Dropout rate. It can effectively prevent overfitting. Set dropout rate to 0.2-0.6 for each experiment. The results of the experiment are shown in Figure 3. On Criteo, the AUC and Logloss reach the optimal value when the dropout rate is 0.5. On Movielens-1M, the AUC and Logloss reach the optimal value when the dropout rate is 0.3. Because Criteo has a wider variety of features than Movielens-1M, resulting in a different degree of optimization.
4.3.3. Activation function. It makes the data linearly separable by adding nonlinear factors. It can solve more complex problems and brings the possibility of the backpropagation of neural networks. Therefore, Relu, Sigmoid, and Tanh were selected to conduct the experiments separately. The results of the experiment are shown in Figure 4. When the activation function is Relu, AUC and Logloss achieve the best performance on the two datasets. Therefore, Relu is the most appropriate activation function in the SELNN.

4.4. Model performance comparison

4.4.1. Parameter settings. PyTorch implemented the experiments. In the hyper-parameter setting, the Embedding size is uniformly set as 15, and Batch Size is uniformly set as 256. For the model with DNN, the number of hidden layers is set as 3, the number of neurons in each hidden layer is set as 128, and the activation function is selected as Relu. The dropout rate is set as 0.5. The optimization is performed with Adam optimizer, and the learning rate is set at $10^{-4}$.

4.4.2. Experimental results and analysis. To validate the effectiveness of SELNN, several traditional and deep learning models have experimented on two datasets. The results are shown in Table 1.
Table 1 Performance of different CTR models on Criteo and Movielens-1M datasets.

| Model | Criteo       | Movielens-1M |       |
|-------|--------------|--------------|-------|
|       | AUC          | Logloss      | AUC   | Logloss |
| LR    | 0.7698       | 0.4974       | 0.8282| 0.3946  |
| FM    | 0.7833       | 0.4772       | 0.8425| 0.3725  |
| PNN   | 0.8009       | 0.4682       | 0.8609| 0.3519  |
| NFM   | 0.7903       | 0.4593       | 0.8589| 0.3485  |
| AFM   | 0.7962       | 0.4541       | 0.8602| 0.3442  |
| DeepFM| 0.8072       | 0.4474       | 0.8627| 0.3413  |
| DCN   | 0.8083       | 0.4453       | 0.8679| 0.3394  |
| SELNN | **0.8146**   | **0.4352**   | **0.8736**| **0.3357** |

On the Criteo dataset, the SELNN improved the AUC values by 4.48% and 3.13% and reduced the Logloss values by 6.22% and 4.2%, respectively, compared to the traditional models LR and FM. Compared with the models PNN, NFM, AFM, DeepFM, and DCN using deep learning methods, the AUC values improved by 1.37%, 2.43%, 1.84%, 0.74%, and 0.63%, respectively, and the Logloss values decreased by 3.3%, 2.41%, 1.89%, 1.22%, and 1.01%, respectively.

On the Movielens-1M dataset, the SELNN improved the AUC values by 4.54% and 3.11% and reduced the Logloss values by 5.89% and 3.68%, respectively, compared to the traditional models LR and FM. Compared with the models PNN, NFM, AFM, DeepFM, and DCN using deep learning methods, the AUC values improved by 1.27%, 1.47%, 1.34%, 1.09%, and 0.57%, respectively, and the Logloss values decreased by 1.62%, 1.28%, 0.85%, 0.56%, and 0.37%, respectively.

In the field of click-through rate forecasting, an increase of 1‰ in AUC is often considered significant. In the above experiments, SELNN achieved the best performance on the two datasets. On the one hand, SENet is used to measure the importance of sparse features, which helps to improve the discriminability of features. On the other hand, adding a logarithmic conversion structure before the DNN can automatically generate different orders of differentiated intersection features and the corresponding weights.

5. Conclusion

To improve the accuracy of CTR prediction, the SELNN proposed in this paper improves the accuracy of CTR prediction utilizing attention mechanism and logarithmic conversion. The improved SENet in this model dynamically learns the importance of features and suppresses the influence of low-frequency and invalid features. CIN can learn higher-order display feature interactions. Moreover, LCN integration with logarithmic conversion structure, and combined with feedforward neural networks to learn the implicit feature interactions and make predictions.

The main focus of this model is on feature engineering, and it lacks the mining of users' history behavior. More effective ways to capture users' interests will be explored in the future to increase the model's effectiveness.

References

[1] CHAPELLE O, MANAVOGLU E, ROSALES R. Simple and scalable response prediction for display advertising [J]. ACM Transactions on Intelligent Systems and Technology (TIST), 2014, 5(4): 1-34.
[2] RENDLE S. Factorization machines with libfm [J]. ACM Transactions on Intelligent Systems and Technology (TIST), 2012, 3(3): 1-22.

[3] QU Y, CAI H, REN K, et al. Product-based neural networks for user response prediction; proceedings of the 2016 IEEE 16th International Conference on Data Mining (ICDM), F, 2016 [C]. IEEE.

[4] XIAO J, YE H, HE X, et al. Attentional factorization machines: Learning the weight of feature interactions via attention networks [J]. arXiv preprint arXiv:170804617, 2017.

[5] WANG R, FU B, FU G, et al. Deep & cross network for ad click predictions [M]. Proceedings of the ADKDD'17. 2017: 1-7.

[6] GUO H, TANG R, YE Y, et al. DeepFM: a factorization-machine based neural network for CTR prediction [J]. arXiv preprint arXiv:170304247, 2017.

[7] WANG X, DONG H, HAN S. Click-Through Rate Prediction Combining Mutual Information Feature Weighting and Feature Interaction [J]. IEEE Access, 2020, 8: 207216-25.

[8] CHENG H-T, KOC L, HARMSEN J, et al. Wide & deep learning for recommender systems; proceedings of the Proceedings of the 1st workshop on deep learning for recommender systems, F, 2016 [C].

[9] YAN C, CHEN Y, WAN Y, et al. Modeling low-and high-order feature interactions with FM and self-attention network [J]. Applied Intelligence, 2021, 51(6): 3189-201.

[10] HUANG T, ZHANG Z, ZHANG J. FiBiNET: combining feature importance and bilinear feature interaction for click-through rate prediction; proceedings of the Proceedings of the 13th ACM Conference on Recommender Systems, F, 2019 [C].

[11] LIAN J, ZHOU X, ZHANG F, et al. xdeepfm: Combining explicit and implicit feature interactions for recommender systems; proceedings of the Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, F, 2018 [C].