Double Cross & Deep Network for News Recommendation

Zhihong Yang[1,a], Yuewei Wu[1,b,*], Muqing Wu[1,c] and Yulong Wang[1,d]

[1]*Beijing University of Posts and Telecommunications, Beijing, China*
[a]*yzh@bupt.edu.cn,* [b]*wuyuewei@bupt.edu.cn,* [c]*wumuqing@bupt.edu.cn,* [d]*wangyulong@ebupt.edu.cn*
**corresponding author*

ABSTRACT
News recommendation algorithms are widely used in many Internet products that people use. With the increase of commercial value, the research on various recommendation algorithms has become more and more interesting. This paper proposes the Double Cross & Deep Network (DCDN) algorithm, which is used in news recommendation. On the basis of the DCN network, the features of "relevant articles" involved in the field of news recommendation are separately extracted, and high-level intersections are made with user information and seed information, respectively. The parameters of the two Cross Network and Deep Network of the DCDN network are independent, and users can change the parameters according to the predicted demand. When the number of layers of the Cross Network is increased, the relevance of the recommendation can be increased, while the equivalent number of layers of the Deep Network can increase the diversity of recommendations. Experiments show that compared with DCN networks, DCDN networks have better parameter performance and faster model operation.

Keywords: *neural networks, feature crossing, deep learning, news recommendation*

1. INTRODUCTION

As the Internet continues to grow in life, people are gradually relying on the Internet to complete activities in their lives. For example, using the Internet to shop, browse news, check the weather, book a restaurant, etc. What followed was the commercial interest of users clicking on links in Internet products. For a large Internet platform, a 0.1% increase in click-through rate can result in tens of thousands of advertising profits or sales profits. Therefore, Click-through rate (CTR) prediction has become a research hotspot in recent years [2][3][4]. On the sales platform, researchers are committed to discovering users' interests in goods, and try to recommend the products that users are most likely to purchase to users to browse to increase sales profits [7]. In the field of news recommendation [5][6], researchers are committed to discovering news that users like to browse, and trying to recommend news that users are most likely to click to increase the user retention time and user click rate, thereby achieving higher advertising profits.

In the field of CTR prediction, identifying effective features is the key to predicting success. However, since the recommendation system in Internet products is mostly used in scenes where data is discrete and classified, resulting in a large and sparse feature space, this is a challenge for feature exploration [8]. Many systems use linear models [1] for identification, such as logistic regression. Although linear models are simple, easy to interpret, and easy to expand, their ability to express is limited and predictive power exists in the ceiling. Therefore, we need to introduce a nonlinear network to better express the cross-features.

Because of the news recommendation, the relevance of relevant news features and user features, the relevance of related news features and the original seed news features are closely related to the click rate. Therefore, we introduced the relationship between the news features and other features by introducing the Double Cross Network. Use Cross Network to most effectively improve the relevance of recommended news to original seed news. At the same time, if the network pays too much attention to the association between news features, although the relevance of the recommended articles can be improved, the diversity of the recommended articles will be largely lost [9][10]. To some extent, the diversity of recommendations can also effectively extend the user's stay time and give users a better experience. Therefore, we also joined Deep Network to adjust the diversity of recommended network results to maximize the recommendation.

1.1. Related Work

As the recommendation algorithm is applied to more and more products, which brings huge benefits, researchers have made many attempts and improvements for the recommendation algorithm. In the earliest algorithms in the recommended field, people use Collaborative Filtering Algorithms [5] to filter related items, articles, and information. However, with the increasing amount of information and features, new challenges such as new user issues and sparseness issues, the effects of collaborative filtering algorithms cannot continue to meet expectations.

In order to solve the problem of feature intersection, the researchers proposed Factor Machine (FM) [11][12]. The proposal of FM solves the problem of feature combination. By combining the features two by two, cross-item features are introduced to improve the effect of the model. At the same time, by introducing a hidden vector, the parameter estimation of the feature is completed, thereby solving the high dimensional disaster. In later studies, researchers also proposed Field-aware Factorization Machines (FFM)[13]. FFM is an upgraded version of FM that introduces the field

Copyright © 2020 The Authors. Published by Atlantis Press SARL.
This is an open access article distributed under the CC BY-NC 4.0 license -http://creativecommons.org/licenses/by-nc/4.0/.

concept. FFM classifies features of the same nature into the same field, which is equivalent to splitting the already subdivided features in FM to perform a two-class model of feature combination.

In order to solve the high-order combination feature problem, the researchers also proposed the Wide & Deep [14] and Deep FM [15] models. The Wide & Deep model treats Deep Network in parallel with the Logistics Regression (LR), and the two parts of the network each have independent inputs. The Deep FM model further changes the LR in the Wide & Deep model to FM parallel processing, which is responsible for the extraction of low-order features and the extraction of higher-order features, respectively. At the same time, the two parts of the network share the same input information. The Deep FM model is improved on the basis of Wide & Deep. It does not require pre-training FM to obtain hidden vectors, and does not require artificial feature engineering. It can simultaneously learn low-level and high-order combination features. The FM module and the Deep Network module share the Feature Embedding section for faster training and more precise training.

Recently, as the recommendation algorithm is applied to major companies, the optimization of model complexity has become more and more important. The Deep & Cross Network (DCN) [16] came into being. The DCN model automatically learns feature intersections for the input of sparse and dense. Effective feature intersections on bounded degrees can be effectively captured without the need for artificial feature engineering or exhaustive searching, which effectively reduces the computational cost.

1.2. *Contributions and Innovations*

The main innovations of this paper are as follows:

- Focusing on the hotspot field of news recommendation, this paper extracts four typical characteristics of user characteristics, seed news features, related news features, and context features for key learning. The relevant news features are combined with user features and seed news features to ensure the relevance of the recommended articles.
- Propose a DCDN network. The network uses two Cross Networks in conjunction with a Deep Network subnet. In the Cross-Network section, the network extracts high-order intersection features between relevant news and user features and seed news features to achieve a good correlation of recommendations. In the Deep Network section, the fully connected network incorporates all features to ensure the diversity of recommendations.
- The experimental part uses multiple data sets to compare the loss and AUC of multiple models horizontally. At the same time, test the memory size occupied by each model and the speed of model training. The DCDN network is evaluated from the engineering perspective to prove the effectiveness of the network in the engineering field.

2. DOUBLE CROSS & DEEP NETWORK (DCDN)

In this section, we will describe the architecture of Double Cross & Deep Network (DCDN) model in detail. The DCDN model begin with the Embedding Layer with variety of features. Then followed by three dependent networks: UR-Cross Network, Deep Network, SR-Cross Network. The final one is Combination Output Layer which combines the three outputs from Networks before. The Complete DCDN network structure is shown as Figure 1.

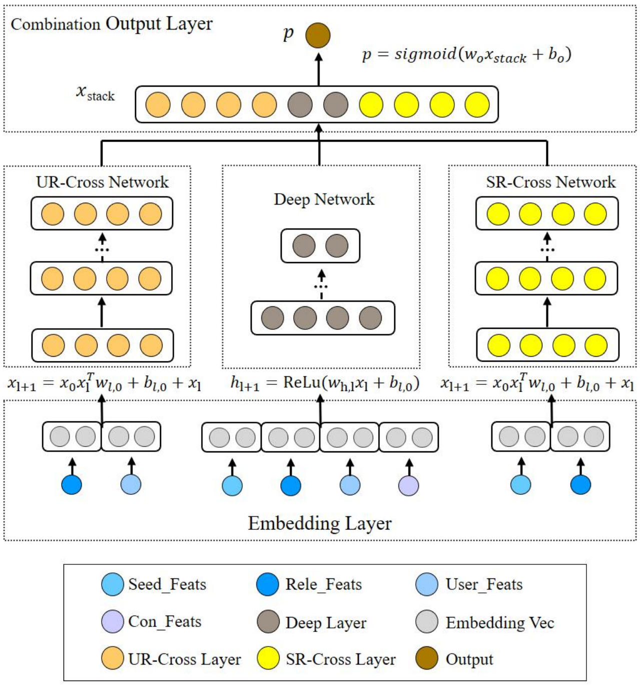

Figure 1 The structure of DCDN

2.1. Embedding Layer

There are four types of features in the news recommendation, including seed news, relevant news, users, and context. We express four types of features as U, R, S, and C. In which, U for User Features, R for Relevant Features, S for Seed Features, C for Context Features. In the embedding layer of the DCDN, we use the matrix W_{embed} to embedding each feature. The algorithm of the Embedding layer is shown in Algorithm 1.

In Algorithm 1, we use the same W_{embed} matrix to perform embedding processing on each feature in the record, where the W matrix is optimized together with other parameters in the network. The algorithm longitudinally combines the vectors obtained by embedding processing into three matrices X, Y, Z, and serves as input to the next layer of the network.

2.2. Double Cross & Deep Network Layer

In the study of recommendation algorithms, feature intersection is one of the key points of the success of the algorithm. Different recommended areas have different feature cross processing modes. Among the news recommendations, the characteristics of relevant news are the most important. The quality of the related news and the degree of relevance are closely related to whether the user ultimately clicks on the news. Through our research, it is found that the relevance of relevant news features and user characteristics, the relevance of related news features and the original seed news features are closely related to the click rate. Improve the relevance of recommended articles and seed news, which can effectively improve the user's click-through rate. At the same time, if the network pays too much attention to the association between news features, although the relevance of the recommended articles can be improved, the diversity of the recommended articles will be largely lost. To some extent, the diversity of recommendations can also effectively extend the user's stay time and give users a better experience.

Therefore, we innovatively attempt to explicitly cross relevant news features and user characteristics, and to explicitly cross relevant news and original seed news features in order to obtain the best relevance. At the same time, join the deep network to enrich the diversity of recommendation results while ensuring relevance. In this way, two crossover networks are formed: UR-Cross Network and SR-Cross Network. The network depths of the two cross-networks are independent of each other, and the number of network layers is l_{ur} and l_{sr}, respectively, which can be independently adjusted. These two networks are used to learn the highly intersecting features of related news features and other features, respectively.

Algorithm 1 Feature Embedding Layer

```
Input:
    Every record $\{u, s, r, c\} \in \{U, S, R, C\}$.    // $\{u, s, r, c\}$ belongs to one sample
    The embedding matrix $W_{embed} \in \Re^{n_e \times n_f}$. In which $n_e, n_f$ are the embedding size and feature size.
Output:
    Embedded Matrix $X, Y, Z$.
Defining function $f_{embed}(t, W_{embed}) = W_{embed} \cdot t$;
for $\{u_i, s_i, r_i, c_i\} \in \{U, S, R, C\}$ do
    $u_{emb} = f_{embed}(u_i, W_{embed})$;
    $r_{emb} = f_{embed}(r_i, W_{embed})$;
    $s_{emb} = f_{embed}(s_i, W_{embed})$;
    $c_{emb} = f_{embed}(c_i, W_{embed})$;
    $X.concat([u_{emb}, r_{emb}])$
    $Y.concat([u_{emb}, r_{emb}, s_{emb}, c_{emb}])$
    $Z.concat([s_{emb}, r_{emb}])$
end for
return $X, Y, Z$;
```

Algorithm 2 Double Cross & Deep Layer

```
Input:
    Number of UR-Cross Network layer $l_{ur}$; Number of SR-Cross Network layer $l_{sr}$;
    Number of Deep Network layer $l_d$; Embedded Matrix $X, Y, Z$.
Output:
    Matrix $X_{l_{ur}}, Y_{l_d}, Z_{l_{sr}}$.
Defining function $f_{cross}(p_l, n) = p_0 p_l w_{l,n} + b_{l,n} + p_l$;
Defining function $f_{deep}(p_l, n) = ReLu(w_{l,n} p_l + b_{l,n})$;
Initialize $X_{l_{ur}}, Y_{l_d}, Z_{l_{sr}} = \emptyset$
for $x_i \in X$ do
    $x_{i,l_0} = x_i$;
    for $l_i \in l_{ur}$ do
        $x_{i,l_i} = f_{cross}(x_{i,l_{i-1}}, l_i)$;
    end for
    $X_{l_{ur}} = X_{l_{ur}}.append(x_{i,l_{ur}})$;
for $y_i \in Y$ do
    $y_{i,l_0} = y_i$;
    for $l_i \in l_d$ do
        $y_{i,l_i} = f_{deep}(y_{i,l_{i-1}}, l_i)$;
    end for
    $Y_{l_d} = Y_{l_d}.append(y_{i,l_d})$;
for $z_i \in Z$ do
    $z_{i,l_0} = z_i$;
    for $l_i \in l_{sr}$ do
        $z_{i,l_i} = f_{cross}(z_{i,l_{i-1}}, l_i)$;
    end for
    $Z_{l_{sr}} = Z_{l_{sr}}.append(z_{i,l_{sr}})$;
    end for
    end for
    end for
return $X_{l_{ur}}, Y_{l_d}, Z_{l_{sr}}$;
```

At the same time, if only the cross-features are used for recommendation, the recommendation coverage will be too narrow, which will affect the accuracy of the recommendation to a certain extent. Therefore, we also added Deep Network for equalization. The deep network is a fully-connected feed-forward neural network, which uses a fully connected form to fuse all features. The Deep Network's network depth is independent of the two Cross Networks, and the depth can be adjusted individually with a network depth of l_d.

Finally, the overall algorithm of the DCDN's Network layer is shown in Algorithm 2.

2.3. Combination and Output Layer

The Combination Output Layer is mainly used to link the vectors output by the network layer and calculate the probability values. We directly concat the vectors output by the three networks, and then use the sigmoid function to calculate the probability value. In the network training process, we use log loss with a regularization term shown as Formula (1).

$$loss = -\frac{1}{N}\sum_{i=1}^{N} y_i \log(p_i) + (1-y_i)\log(1-p_i) + \lambda \sum_{l} ||w_l||^2$$

The overall algorithm for the Combination and Output Layer is shown in Algorithm 3.

Algorithm 3 Combination and Output Layer

```
Input:
    The out put of Network $X_{l_{ur}}, Y_{l_d}, Z_{l_{sr}}$.
Output:
    Predicted probability $p_i$.
for $x_{i,l_{ur}} \in X_{l_{ur}}, y_{i,l_d} \in Y_{l_d}, z_{i,l_{sr}} \in Z_{l_{sr}}$ do
    $q_i = [x_{i,l_{ur}}, y_{i,l_d}, z_{i,l_{sr}}]$;
    $p_i = sigmoid\ (w_i q_i + b_i)$;
end for
return $p_i$;
```

3. EXPERIMENT

3.1. Data Preprocessing

This article conducted experiments on two datasets: Outbrain and RealNews. Outbrain is a public dataset that was publicly provided by Kaggle Competition in 2017. This data set is the browsing and clicking behaviors of relevant news or advertisements recommended by the end of the text on more than 560 websites in the United States from June 14 to June 28, 2016. The data involves 2 billion browsing records of 700 million different users, 87 million recommendation records, 16.9 million click behaviors, and corresponding text and context information, and is included in multiple related files such as page_views, clicks, documents, events, and so on. Text and context information includes publisher, time of publication, topic, category, timestamp, platform, and more. After filtering users with less than 10 page views, the combined sample was used to generate the final sample used in the experiment.

RealNews is real data recommended by client news from a Chinese company. The recommendation scenario is to personally recommend relevant news content for the user at the end of the news page opened by the user. From the two-week online recommendation behavior log, extract the required feature fields, including user portrait features, news features, context features, etc., to generate a recommendation data set. The unexposed data and the number of exposures less than 10 are removed from the hundreds of millions of pulled data, and the experimental samples are finally generated after combined screening. The data amount, feature number and other information of the two data sets are shown in Table 1.

Table 1 Data set basic information

Data Set	Num of Users	Num of samples	Num of feature groups	Num of features
Outbrain	1M	10M	20	0.1M
RealNews	2M	15M	27	11M

3.2. Models for Comparisons

We compare DCDN with four original models: the DCDN model with Factorization Machine (FM), Wide and Deep Model (W&D), Deep Factorization Machine (DeepFM) and Deep and Cross Network (DCN).

FM. The input is raw sparse features.

W&D. The Wide part uses the original sparse features as input. Since the parameters of the W&D model are difficult to adjust, the same parameters and structures as those proposed in the paper are used.

DeepFM. The original sparse features and Embedding layer are the same as DCDN, and the rest of the network structure is set as mentioned in the paper.

DCN. The original input and Embedding layer are the same as the DCDN. The difference is that there is only one Cross Network. The input of this part is the same as the Deep Network part, which is the combination of all Embedding vectors.

3.3. Models Performance

In this section, we first compare the AUC parameters and logloss parameters of different networks on different data sets in the optimal case. After that, we further compare the DCDN with the DCN network line that most closely resembles it. The final conclusion is obtained through the effect difference between the models.

We use TensorFlow to build each network and test it separately in different data sets, so that each model reaches the best state of training. When the model is in the best state, the AUC parameters of each model in each data set are shown in Table 2. The logloss parameters of each model are shown in Table 3.

Table 2 AUC parameters of different models in different data sets

AUC	FM	W&D	DeepFM	DCN	DCDN
Outbrain	0.7474	0.7521	0.7529	0.7532	**0.7547**
RealNews	0.7633	0.7715	0.7731	0.7746	**0.7763**

Table 3. Logloss parameters of different models in different data sets

Logloss	FM	W&D	DeepFM	DCN	DCDN
Outbrain	0.3904	0.3866	0.3865	0.3858	**0.3841**
RealNews	0.3596	0.3547	0.3532	0.3526	**0.3512**

From Table 2 and Table 3, we can see that the DCDN model performs well in both data sets. Compared with the best-performing DCN network, the AUC parameter has improved about 0.2% in both data sets. Compared with the best performing DCN network, the logloss parameter has reduced the performance in both data sets by about 0.4%. For an actual data set, any 0.1% improvement is of great value for practical applications. The parameter improvement of the DCDN model is very considerable.

At the same time, from the perspective of practical applications, we need not only the optimization of model parameters, but also a faster training speed. Model complexity, model training time, and model stability are all important measures in engineering. Therefore, we tested the speed angle of model training for the above five networks. In the test, we used python 2.7.15 version, TensorFlow 1.12.1 version, scikit-learn 0.20.1 version and pandas 0.23.4 version for testing. All models are tested in a single-threaded manner. Under the unified calculation conditions, the number of samples ($\times 10^3$) calculated every second is tested, as shown in Table 4.

Table 4 Model operation speed

Samples ($\times 10^3$)	FM	W&D	DeepFM	DCN	DCDN
Outbrain	10.1	9.7	9.4	9.1	9.3
RealNews	2.7	2.6	2.7	2.6	2.7
Mean	6.4	6.2	6.1	5.9	6.0

From Table 4, we can see that the DCDN network is generally superior to the DCN network in terms of computing efficiency. In terms of network structure, FM has the simplest network structure, followed by W & D and DeepFM networks, and DCN is close to DCDN. Therefore, from the perspective of model complexity, FM, W & D, and DeepFM are bound to have an advantage in terms of speed. However, in the case that the DCDN network greatly improves the model effect, the model speed does not decrease much, which shows that the advantages of DCDN obviously exist. At the same time, we separately compare DCN network and DCDN network. As can be seen from Table 2-4, compared to the DCN network, the parameters of the DCDN network perform better, and at the same time the computing speed is also improved. This fully proves that the DCDN network performs better on this data set.

4. CONCLUSION

This paper mainly proposes the DCDN network, and elaborates the DCDN model in detail from the perspective of the model structure and algorithm of the algorithm. On the basis of the DCN network, we separately extracted the features of "relevant articles" involved in the field of news recommendation, and conducted high-level intersections with user information and seed information. The parameters of the two Cross Network and Deep Network of the DCDN network are independent, and users can change the parameters according to the predicted demand. When the number of layers of the Cross Network is increased, the relevance of the recommendation can be increased, while the equivalent number of layers of the Deep Network can increase the diversity of recommendations. This makes the DCDN network structure more suitable for the research field of news recommendation. We verified the effectiveness of the DCDN network with two datasets, DotBrain and RealNews, respectively. The experimental results show that the DCDN network has greatly improved the prediction effect compared with the FM, W & D, DeepFM, and DCN networks. Compared with the DCN network, the parameters of the DCDN network perform better, and the model operation speed is faster. This fully proves the practicality of the DCDN network.

REFERENCES

[1] Chapelle O, Manavoglu E, Rosales R. Simple and Scalable Response Prediction for Display Advertising[J]. ACM Transactions on Intelligent Systems and Technology, 2014, 5(4):1-34.

[2] Graepel T, Candela J Q, Borchert T, et al. Web-scale bayesian click-through rate prediction for sponsored search advertising in microsoft's bing search engine[C]. Omnipress, 2010.

[3] Zhou G, Zhu X, Song C, et al. Deep interest network for click-through rate prediction[C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, 2018: 1059-1068.

[4] Tuladhar L, Gupta M S. Click through rate prediction system and method: U.S. Patent 8,738,436[P]. 2014-5-27.

[5] Liu J, Dolan P, Pedersen E R. Personalized news recommendation based on click behavior[C]//Proceedings of the 15th international conference on Intelligent user

interfaces. ACM, 2010: 31-40.

[6] IJntema W, Goossen F, Frasincar F, et al. Ontology-based news recommendation[C]// Proceedings of the 2010 EDBT/ICDT Workshops. ACM, 2010: 16.

[7] Zhao X W, Guo Y, He Y, et al. We know what you want to buy: a demographic-based system for product recommendation on microblogs[C]//Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2014: 1935-1944.

[8] Li H, Zhang S, Wang X. A Personalization Recommendation Algorithm for E-Commerce[J]. JSW, 2013, 8(1): 176-183.

[9] Bradley K, Smyth B. Improving recommendation diversity[C]//Proceedings of the Twelfth Irish Conference on Artificial Intelligence and Cognitive Science, Maynooth, Ireland. 2001: 85-94.

[10] Hurley N, Zhang M. Novelty and diversity in top-n recommendation--analysis and evaluation[J]. ACM Transactions on Internet Technology (TOIT), 2011, 10(4): 14.

[11] Rendle S. Factorization Machines[C]// 2011.

[12] Ta A P. Factorization machines with follow-the-regularized-leader for CTR prediction in display advertising[C]// 2015 IEEE International Conference on Big Data (Big Data). IEEE, 2015.

[13] Juan Y, Zhuang Y, Chin W S, et al. Field-aware factorization machines for CTR prediction[C]//Proceedings of the 10th ACM Conference on Recommender Systems. ACM, 2016: 43-50.

[14] Cheng H T, Koc L, Harmsen J, et al. Wide & deep learning for recommender systems[C]//Proceedings of the 1st workshop on deep learning for recommender systems. ACM, 2016: 7-10.

[15] Guo H, Tang R, Ye Y, et al. DeepFM: a factorization-machine based neural network for CTR prediction[J]. arXiv preprint arXiv:1703.04247, 2017.

[16] Wang R, Fu B, Fu G, et al. Deep & cross network for ad click predictions[C]//Proceedings of the ADKDD'17. ACM, 2017: 12.