Accurate and Visual Video recommendation based on Deep Neural Network

Fan Yang
School of Advanced Technology
Xi'an Jiaotong-Liverpool University
Suzhou,CN
yangfan@liverpool.ac.uk

Gangmin Li
School of Computer Science and Technology
University of Bedfordshire
Luton, UK
Gangmin.li@beds.ac.uk

Yong Yue
School of Advanced Technology
Xi'an Jiaotong-Liverpool University
Suzhou,CN
yongyue@xjtlu.edu.cn

Terry Payne
Department of Computer Science
University of Liverpool
Liverpool, UK
T.R.Payne@Liverpool.ac.uk

***Abstract*—Video recommendation is vital for a video platform, which provides its users with videos they may be interested in. In this paper, we integrate users' ratings of videos in the video platform and community and crucial information data such as video category, director/actor, predict users' preference for videos through deep neural network, which could improve the accuracy of personalized recommendation. In addition, we use weighted force-directed Graph to show the relationship among users, videos, directors, and other elements, which could display the visualization of data elements and recommended results. Extensive experiments are conducted on three video datasets, and the experimental results demonstrate that the proposed method is more effective than several other recommendation methods.**

Keywords—Personalized Recommendation, Deep Neural Network, Data Visualization, Video Recommendation

I. Introduction

Nowadays, large amounts of video platforms and live platforms are promoted by explosive growth of mobile internet, and the flow of videos exceeds half of the flow of internet. Video recommendation systems are designed on most large-scale video platforms such as Youtube, IQiyi and Netflix, to provide users with better watching experiences. In the field of recommendation, more and more recommender systems apply deep learning technology and significantly improve the accuracy and recall of recommendation, which makes the recommender system based on deep neural network become a new hot research direction. At the same time, with the advent of the big data era, data visualization has attracted more and more attention. By visually presenting the recommendation results, users can more intuitively accept the content recommended by the system and improve the interpretability of the recommendation system.

However, we found that some video platforms have serious problems: 1. Too much content is occupied by non-personalized recommendations, such as hotlists, recent views, latest videos, advertisements, etc.; 2. Most of the recommended videos have been watched by users, which loses the significance of recommendation; 3. Recommendation results are often unexplainable, which makes users not accept the recommendation. Many video platforms still take video resources as their selling point, and the development of personalized recommendation is far from enough. Relatively, the recommendation of short video platforms is very effective, and they are developing very rapidly relied on personalized recommendation.

We proposed an effective method to realize the accuracy and visualization of video recommendation to solve these practical issues. Our method can classify video recommendation results, for example, according to the recommendations of relevant actors and directors or according to the video type. Besides, our method can recommend similar users, i.e., potential friends, and digitize and visualize the recommendation results to make the recommendation results more credible. Our contributions are summarized as follows:

- We realize the personalized video recommendation through deep neural network, to improve the accuracy and coverage of recommendation;
- We realize the visualization of recommendation results through weighted force-directed Graph algorithm so that it has a more reasonable recommended explanation;
- We perform experiments on real-world data, and the results show that our method outperforms other methods in terms of recall and ndcg evaluation metrics.

II. Related Work

A. Video Recommendation

With the rapid development of the video field, the discussion and research on personalized video recommendation is gradually deepening. Throughout China's video platform, Tencent Video, iQiyi, Douban, etc., all have personalized recommendation systems. Traditional video recommendation algorithms are usually divided into four categories: collaborative filtering recommendation, content-based recommendation, demographic recommendation, and hybrid recommendation. User-based collaborative filtering is to find user groups with similar interests to the target users from users' perspectives [1]. Item-based collaborative filtering [2] from the perspective of items recommends by looking for similar items of items related to the target user. The principle of content-based recommendation is to find similar items and recommend them to users according to the category, label, comment, and other relevant information of items that users have history behavior [3]. Because content-based recommendation only needs to calculate the similarity according to the item features and does not need other user information, there is no cold start and new item recommendation issue. The recommendation based on demography is to recommend according to users' basic information. Generally, the user's basic information includes age, occupation, nationality, gender and home address. Users are divided into different categories according to their basic information. Users in the same category are similar, and multiple categories are more similar [4]. With the increase of the number and types of items, users' requirements for personalized recommendation are more diversified and complicated. Using a single recommendation algorithm cannot meet people's diversified needs. To solve the limitations of a single algorithm, researchers propose a hybrid recommendation model [5]. Its basic idea is to integrate the advantages of various recommendation algorithms, develop their strengths and avoid their weaknesses, obtain the optimal recommendation and deal with complex recommendation tasks.

B. Recommendation Visualization

In recent years, the research on recommendation visualization used visualization technology to display and adjust recommendation results. We divide it into recommendation visual layout and recommendation visual interaction. The recommendation visual layout focuses on the presentation of the recommendation results, which can be divided into node-link layout (the recommendation process is visualized in the form of node-link graph), collection based layout (it is convenient to view the author attributes at the higher level of the recommendation results and the relationship between them), radial layout (a representation of the inclusion relationship of the label), table layout (increasing the user's understanding of the recommendation process and system trust) and scatter diagram (visualizing the feature space). Visual interaction technology is also mainly used to present recommendation results and understand the recommendation process. We summarized it into three aspects: the adjustment of item attributes, user profile information, and the selection and mixing of recommendation algorithms.

C. Recommendation based on Deep Neural Network

Recommendation based on DNN is a model to realize recommendation tasks based on deep neural network architecture. The typical recommendation based on DNN can be divided into recommendation based on self-encoder, recommendation based on Factorization Machines, recommendation based on CNN, and recommendation based on RNN. Factorization Machines (FM) is a model based on matrix factorization, which was proposed by Rendele et al. [6-8]. It can combine feature data in linear time to obtain cross-feature knowledge and overcome data sparsity to a certain extent. FM is limited to two aspects. One is that it only combines different features without considering multiple features in the same field; Second, it is challenging to produce high-order features. Therefore, Juan et al. [9] proposed Field-aware Factorization Machine, which further optimized FM to strengthen the relationship between a class of features in the field; Blondel et al. [10] proposed Higher-order FM, which optimized the crossover ability of FM to high-order features. However, these models are much inferior to deep neural networks in capturing potential high-order knowledge in data. Researchers try to combine feature engineering with neural networks. Therefore, Shan et al. [11] proposed Deep Crossing to enable neural networks to learn feature combination autonomously to realize feature engineering automation; Cheng et al. [12] proposed Wide&Deep, which realizes the prediction by taking the cross feature as the input of the linear model and jointly training the linear model with the depth neural network; Wang et al. [13] proposed Deep&Cross Network to optimize the feature selection and combination methods of Deep Crossing and Wide&Deep by building a multi-layer cross network; Guo et al. [14] proposed DeepFM to share the same input for FM and DNN, in which FM learns low-order knowledge and DNN learns high-order knowledge; He et al. [15] proposed the Neural Factorization Machine (NFM), which takes FM as the entrance, and then obtains the second-order feature interaction knowledge from the low-order features through the Bi-interaction layer.

III. METHODOLOGY

The framework of our method is shown in Fig. 1. It consists of the following three modules: Item set visualization, recommendation engine, recommendation results visualization, and feedback. The item set visualization visualizes the recommended item set and assists the user in exploring and generate preference data, which is input into the recommendation engine. After processing, the recommendation engine outputs intermediate data and results to the recommendation results visualization, which interprets and visually presents the recommendation results. At the same time, it feeds back the data interaction between the user and the recommendation results into the recommendation engine which can enrich user data and optimize the next round of recommendation results.

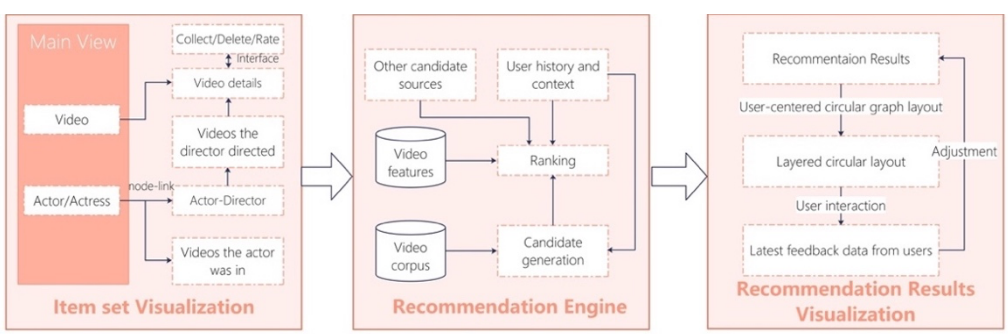

Fig. 1 Overall framework of accurate and visual video recommendation

A. *Item set visualization*

We use the "Item multi-attribute visualized interaction" design pattern to visualize the video set. The primary view is video view and actor relationship view. In the video view, we use the year attribute filtering. After the user selects the interval of the release year and the number of explorations, the system obtains the data in the interval and generates a visual layout. The actor relationship view generates the layout according to the number of people selected by the user and provides the edge filtering method of relationship weight to display critical information. Displays a dynamic view when the user interacts with a node in the home view. The dynamic view includes a visual view of the actor-director relationship corresponding to the actor when double-clicking the actor node; Click the actor-director node to display the videos played by the actor or directed by the director, which can be sorted by score or by year; Click the video node to display the details of this video, including name, poster, etc. Users can perform interactive operations such as collection, deletion, and scoring on the card view.

We adopts Douban-Movie dataset, which has complete movie attribute information. According to video market data, the influence of famous actors has a dominant impact on the film box office. Due to the huge volume of director and actor data sets, we screen actor data. Considering that the audience's memory of famous actors generally refers to the starring of well-known and highly praised films, it is analyzed that "well-known" can correspond to the number of audience comments of the video, "high praise" corresponds to the rating of the video, and whether "starring" corresponds to the ranking of actors in the video. We set the actors whose ranking is less than 6 as the starring actors of the video, We use formula (4) to calculate influence weight of actor A_i:

$$w_{A_i} = \sum_{m \in M_{A_i}} R_m \times N_m \tag{1}$$

Where M_{A_i} is a collection of video played by actor A_i, R_m is the average rating of video m, N_m is the number of ratings on video m. Then, the relationship weights actor-actor, actor-director are calculated respectively.

B. *Recommendation engine based on DNN*

The principle of deep recommender system is to provide various original data of users and items to the input layer, learn and extract the hidden features of users and items through neural network in the hidden layer, and finally realize user and item recommendation through learning implicit representation [16-17]. The architecture of recommendation based on deep neural network is shown in Fig. 2. Firstly, the history data such as user browsing and search items are mapped into vectors. The fixed-length representation is obtained by averaging the vectors; input user profile features to optimize the effect of personalized recommendation. Secondly, all input embeddings are concatenated into the same vector, and the concatenated vector is input to the activation function. Finally, the output is trained by Softmax for classification. The similarity between the training features and the source items is calculated, obtaining the Top N items with the highest similarity as the recommendation results.

From users' perspective, their watch history, search tokens, geographic information, gender, age, watching time, etc., are used as embeddings to input X_u. Then the implicit representation Y_u is learned and output through the deep neural network. From the perspective of items, the item's context, label, and type are used as embeddings input X_i, and the implicit representation Y_i is learned through the deep neural network. The DNN model from the user perspective is $f_u(x_u, w_u)$, and the DNN model from the i^{th} item perspective is $f_i(x_i, w_i)$. If there are M samples $\{(x_{u,j}, x_{a,j})\}, 0 \leqslant j \leqslant M$, $(x_{u,j}, x_{a,j})$ is the interaction between user u and item a, and the relevant interaction records of user and item are used for parameter adjustment learning:

$$\arg_{w_u, w_1, \cdots, w_v} \max \sum_{j=1}^{M} \frac{e^{\cos\left(f_u(x_{u,j}, w_u), f_a(x_{a,j}, w_a)\right)}}{\sum_{1 \leqslant a \leqslant N} e^{\cos\left(f_u(x_{u,j}, w_u), f_a(x_{a,j}, w_a)\right)}} \tag{2}$$

The user implicit representation Y_u, and item implicit representation Y_i, obtained after model training, are used to calculate the similarity and ranking between users and items in the implicit space, select the K items with high similarity, and collaborative filtering of the source database to achieve accurate and personalized recommendation.

The MLP layer uses the common tower design. The bottom layer is the widest, and the number of neurons in each layer is halved until the softmax input layer is 256 dimensions (1024ReLU→512ReLU→256ReLU). The input data of the full connection layer (hidden layer) is the embeddings of users and items. Through ReLU activation function of the hidden layer, the vector u_i is the hidden feature vector of the user $user_i$. Similarly, the hidden feature vector of the item $item_j$ is v_j. The calculation is as follows:

$$u_i = \mathrm{ReLU}\left(w_{user\ i} P_{user\ i} + b_{user\ i}\right) \tag{3}$$

$$v_j = \mathrm{ReLU}\left(w_{item\ j} P_{item\ j} + b_{item\ j}\right) \tag{4}$$

Where $u_i, v_j \in R^m$, w_{user_i} and w_{item_j} respectively represent the weights of user and item in the full connection layer; b_{user_i} and b_{user_j} are the corresponding bias respectively; P_t is the largest term of dimensionality reduction after inputting embeddings, $t \in [user_i, item_j]$; $\mathrm{ReLU}(x) = \max(0, x)$.

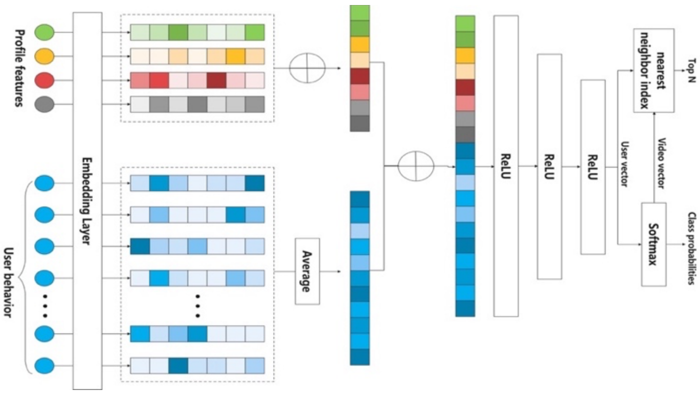

Fig. 2 The architecture of recommendation based on deep neural network

C. Recommendation Results Visualization

The recommendation results visualization module mainly uses the user-centered force-directed graph layout algorithm to generate a layered circular visual view with the output by the recommendation engine. Users can see the recommendation results and recommendation path. In addition, users can interact with views, such as click, collect, rate, etc. At the same time, it also provides explanations of a variety of recommendation results, such as recommendations that users may like, recommendations based on similar movies, recommendations based on similar users, etc.

In our work, we adopt weighted force-directed algorithm to implement recommendation results visualization. Weighted force-directed algorithm is an improvement of the classical Fruchterman-Reingold algorithm, because FR does not consider the weight of each edge, and the weight of the edge can effectively reflect the relationship between nodes. For example, in the visualization of recommendation results, we tend to cluster similar information, and the distance between nodes is used to represent its similarity. Repulsion and attraction in Weighted force-directed algorithm are defined as follows:

$$F_r(u,v) = (rc * k)^2 / d(u,v) \tag{5}$$

$$F_a(u,v) = d(u,v)^2 * w(u,v)/k \tag{6}$$

Where rc is the constant of repulsion factor and $0 < rc \leq 1$; $d(u,v) = \|p_u - p_v\|$ is the geometric distance between node u and node v; k is the balancing factor, which indicates the side length of the square region where each node is located if evenly distribute the $|v|$ nodes in graph G in the region of $w * h$; $w(u,v)$ is the weight of the edge (u,v) between node u and node v.

We can see that when calculating the attraction between node u and v, we should not only consider the Euclidean distance between nodes, but also consider the actual relationship value between nodes. We have proposed to normalize $w(u,v)$ to make its value between $(0,1)$. The reason for this is that if simply multiply the weight, the repulsion will be much greater than the attraction, resulting in the length of the edge being infinitely enlarged. The layout area is limited, and most of the points will be arranged to the edge, resulting in the poor effect of the algorithm.

The weighted force-directed algorithm refers to the physical model, inherits the characteristics of FR algorithm, which is easy to understand and has high freedom, and can be well applied to visualization.

IV. EXPERIMENTS

We provide empirical results to demonstrate the effectiveness of our proposed method. The experiments are designed to answer the following research questions:

- RQ1: How does our method perform on the accuracy of recommendation?
- RQ2: Can our method visualize recommendation results and give an intuitive impression of explanation?

A. Experimental Settings

1) Datasets

a) MovieLens: This dataset is about movie ratings and has been widely used to evaluate recommendation algorithms. We use ML-1m containing 1 million rating records, respectively. We extract interaction records from rating data, items from "movie name", and users from "user id".

b) Douban Movies: This dataset is about movie ratings collected from Douban website. In the dataset, we use friend relationships, rating data, and genres of movies as categories.

c) Amazon Movies & TV: This dataset contains product reviews and metadata from Amazon. It includes reviews (ratings, text, helpfulness votes), product metadata (descriptions, category information, price, brand, and image features), and links (also viewed/also bought graphs).

2) Alternative Baselines

a) MF (Matrix Factorization) [18]: only considers the user-item interactions while leaving KG untouched. Technically, it uses ID embeddings of users and items to predict.

b) BPR (Bayesian Personalized Ranking) [19]: learns personalized ranking from implicit feedback.

c) FPMC [7]: captures users' general preference and a first-order MC to predict the user's following action.

d) PRP (PageRank with Priors) [20]: integrate the user-item relationship and other heterogeneous auxiliary information into a unified isomorphism diagram, and then PageRank each user with a personalized initial probability distribution.

3) Evaluation Metrics

To evaluate the recommendation performance of our method, we use two evaluation metrics recall@K and ndcg@K. The first metric evaluates the fraction of ground truth items that have been retrieved over the total amount of ground truth items, while the second metric is Normalized Discounted Cumulative Gain which takes the exact position of the correctly recommended items into account. The larger the values of both recall and ndcg metrics, the better the performance.

$$\begin{aligned} \mathrm{NDCG@}\,k &= \frac{1}{Z}\,\mathrm{DCG@}k \\ &= \frac{1}{Z}\frac{1}{|\mathcal{U}|}\sum_{u\in\mathcal{U}}\sum_{i=1}^{k}\frac{\delta(r_{u,i}\in\mathcal{T}_u)}{\log_2(i+1)}, \end{aligned} \tag{5}$$

where $r_{u,i}$ is the k^{th} item recommended for user u. Z is a normalization constant, which is the maximum possible value of DCG@k.

4) *Parameter Settings*

The weights of user and item eigenvalues are 1 and 0.5 respectively, the model learning rate is 0.00065, and the regularization of user and item implicit features is $\lambda_{user} = \lambda_{item} = \lambda$， The number of deep neural network neurons is 1026. The three real data sets are randomly divided into 80% Trainset and 20% Testset, and there is no intersection between them. At the same time, 20% data in Testset are randomly used for verification to adjust model parameters.

B. *Performance Comparison(RQ1)*

We begin with the comparison concerning recall@20, recall@50, ndcg@20 and ndcg@50. Table 1 shows the empirical results, with percent Imp. denoting the relative improvements of the top-performing technique over the strongest baselines. We can find that our method consistently outperforms all baselines across three datasets in terms of all measures. More specifically, it achieves significant improvements over the strongest baselines concerning ndcg@50 by 9.19%, 13.8%, and 4.41% in MovieLens, Douban-Movie, and Amazon Movies&TV, respectively. Our model's logic and efficacy are demonstrated in this way.

TABLE I. RECOMMENDATION PERFORMANCE COMPARISON

Dataset	Metric	MF	BPR	FPMC	PRP	Ours	%Imp
MovieLens	recall@20	0.0612	0.1010	0.1357	0.1209	0.1415	4.27
	recall@50	0.1332	0.1351	0.1489	0.1320	0.1546	3.83
	ndcg@20	0.0417	0.0560	0.0812	0.0539	0.0878	8.13
	ndcg@50	0.0621	0.0771	0.0903	0.0721	0.0986	9.19
Douban-Movie	recall@20	0.0805	0.1017	0.1284	0.1262	0.1287	0.23
	recall@50	0.1389	0.1543	0.1651	0.1520	0.1703	3.15
	ndcg@20	0.0477	0.0544	0.0745	0.0798	0.0841	5.39
	ndcg@50	0.0665	0.0705	0.0883	0.0862	0.1005	13.8
Amazon Movies&TV	recall@20	0.0912	0.1082	0.0960	0.0982	0.1110	2.59
	recall@50	0.1271	0.1345	0.1402	0.1381	0.1452	3.57
	ndcg@20	0.0408	0.0502	0.0572	0.0550	0.0587	2.62
	ndcg@50	0.0642	0.0757	0.0794	0.0763	0.0829	4.41

C. *Visualization of Recommended Results(RQ2)*

We sample ten users from the datasets and visualize the items between one selected category and the observed items belonging to each user's category. Then the visualization is shown as Fig. 3(1)(2), in which we can observe the interaction of user and item as well as the path of recommendation. The recommendation results display on the views, and we can interact with views, such as click, collect, rate, etc. we can also see the explanations of recommendation results, such as recommendations that users may like, recommendations based on similar movies, recommendations based on similar users, etc.

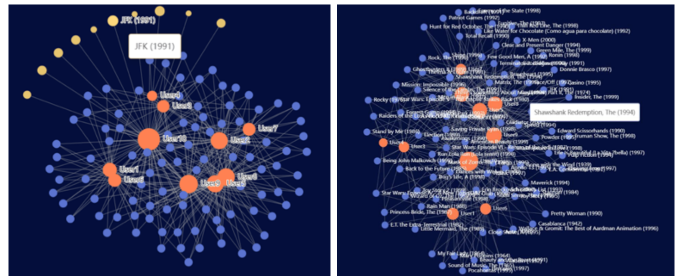

Fig. 3(1)(2) Visualization of item set and recommendation visualization

V. CONCLUSION

This paper uses video data to study personalized recommendation methods and recommendation visualization methods. Combined with deep neural network and data visualization technology, we realize the complete process from data input, recommendation prediction to result in visualization. Firstly, based on the historical interaction of users on the video platform and the critical information data such as the category, director and actor of the video itself, a deep recommendation architecture is proposed to predict users' video preference which could be called personalized recommendation. Then, based on the force-oriented graph layout, a visual presentation method of recommendation results is proposed; Finally, a series of experiments show the effectiveness of our proposed method.

ACKNOWLEDGMENT

This work is supported by the Basic Public Welfare Research Project of Zhejiang (LGF20G020001)， Key Lab of Film and TV Media Technology of Zhejiang Province (No.2020E10015)，and the AI University Research Centre (AI-URC) through the XJTLU Key Program Special Fund (KSF-A-17).

REFERENCES

[1] Zhang X.C, Zhou Z.H, et al. Improved collaborative filtering algorithm based on association rules [J] Journal of Chongqing University of Technology (NATURAL SCIENCE), 2019,33 (3): 161-168

[2] Salam Salloum and Dananjaya Rajamanthri, "Implementation and Evaluation of Movie Recommender Systems Using Collaborative Filtering," Journal of Advances in Information Technology, Vol. 12, No. 3, pp. 189-196, August 2021

[3] Thunder, Zhu.M. Application of emotion analysis in film recommendation system [J] Computer engineering and application, 2016,52 (10): 59-63107

[4] Jiyeon Kim, Youngchang Kim, Hyesun Suh, and Jongjin Jung, "Diversity of Recommendation with Considering Data Similarity among Different Types of Contents," Journal of Advances in Information Technology, Vol. 7, No. 2, pp. 76-80, May, 2016

[5] Subhash K. Shinde and Uday V. Kulkarni, "Hybrid Personalized Recommender System Using Fast K-medoids Clustering Algorithm,"

Journal of Advances in Information Technology, Vol. 2, No. 3, pp. 152-158, August, 2011

[6] RENDLE S. Factorization machines[C]//Proceedings of 2010 IEEE International Conference on Data Mining. Sydney: IEEE, 2010: 995

[7] RENDLE S. Factorization machines with libFM[J]. ACM Transactions on Intelligent Systems and Technology, 2012, 3(3):1

[8] RENDLE S, GANTNER Z, FREUDENTHALER C, et al. Fast context-aware recommendations with factorization machines[C]//Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval. Beijing: ACM, 2011: 635

[9] JUAN Yuchin, ZHUANG Yong, CHIN Weisheng, et al. Field-aware factorization machines for CTR prediction[C]//Proceedings of the 10th ACM Conference on Recommender Systems. Boston: ACM, 2016:43

[10] BLONDEL M, FUJINO A, UEDA N, et al. Higher-order factorization machines[C]//Proceedings of the 30th International Conference on Neural Information Processing Systems. Barcelona: Curran Associates Inc., 2016:3359

[11] SHAN Ying, HOENS T R, JIAO Jiao, et al. Deep crossing: Webscale modeling without manually crafted combinatorial features[C]//Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. San Francisco: ACM, 2016:255

[12] CHENG Hengtze, KOC L, HARMSEN J, et al. Wide&deep learning for recommender systems[C]//Proceedings of the 1st workshop on deep learning for recommender systems. Boston: ACM, 2016:7

[13] WANG Ruoxi, FU Bin, FU Gang, et al. Deep&cross network for ad click predictions[C]//Proceedings of the ADKDD 2017. Halifax: ACM, 2017: 12.

[14] GUO Huifeng, TANG Ruiming, YE Yunming, et al. DeepFM: A factorization-machine based neural network for CTR prediction[C]//Proceedings of the 26th International Joint Conference on Artificial Intelligence. Melbourne: AAAI Press, 2017:1725

[15] HE Xiangnan, CHUA Tatseng. Neural factorization machines for sparse predictive analytics[C]//Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval. Tokyo: ACM, 2017:355

[16] Covington P, Adams J, Sargin E. Deep Neural Networks for YouTube Recommendations[C]//Acm Conference on Recommender Systems. ACM, 2016:191-198.

[17] Word Yunfei, Li Yeli, sun Huayan Research on Personalized Recommendation System Based on deep neural network [J] Electronic technology application, 2019, 45 (1): 6

[18] L. Baltrunas, B. Ludwig, and F. Ricci. Matrix factorization techniques for context aware recommendation[C]. In Proceedings of the fifth ACM conference on Recommender systems, 2011: 301–304.

[19] RENDLE S, FREUDENTHALER C, GANTNER Z, et al. BPR: bayesian personalized ranking from implicit feedback[C]. Proceedings of The 25th Conference on Uncertainty in Artificial Intelligence, 2009: 452–461.

[20] NGUYEN P, TOMEO P, DI NOIA T, et al. An evaluation of SimRank and personalized PageRank to build a recommender system for the web of data[C]. Proceedings of the 24th International Conference on World Wide Web, 2015: 1477–1482.