



## Applied Mathematics and Nonlinear Sciences

<https://www.sciendo.com>

## Agricultural Product Recommendation Model based on BMF

Fucheng Wan<sup>1</sup>, Dengyun Zhu<sup>2</sup>, Xiangzhen He<sup>1†</sup>, Qi Guo<sup>1</sup>, Dongjiao Zhang<sup>1</sup>, Zhenyang Ren<sup>1</sup>, Yuxiang Du<sup>3</sup><sup>1</sup>Key Laboratory of China's Ethnic Languages and Information Technology of Ministry of Education Northwest Minzu University, Lanzhou, China<sup>2</sup>Northwest Minzu University, Lanzhou, China<sup>3</sup>Deakin University, Geelong, Australia

## Submission Info

Communicated by Juan Luis García Guirao

Received July 10th 2020

Accepted August 14th 2020

Available online December 21st 2020

## Abstract

In this article, based on the collaborative deep learning (CDL) and convolutional matrix factorisation (ConvMF), the language model BERT is used to replace the traditional word vector construction method, and the bidirectional long–short time memory network Bi-LSTM is used to construct an improved collaborative filtering model BMF, which not only solves the phenomenon of ‘polysemy’, but also alleviates the problem of sparse scoring matrix data. Experiments show that the proposed model is effective and superior to CDL and ConvMF. The trained MSE value is 1.031, which is 9.7% lower than ConvMF.

**Keywords:** Recommendation system, BERT model, Bi-LSTM cost**AMS 2010 codes:** E4395

## 1 Introduction

Along with the popularisation of the internet, e-commerce has gradually integrated into people's lives and become an integral part. With the increasingly fierce competition among major e-commerce companies, it is very necessary to build an excellent recommendation system. Amazon, as a giant in the foreign e-commerce industry, took the lead in using the recommendation system to provide services for users, which not only brought huge economic benefits in a short period of time, but also trained many loyal users. The success of Amazon has led to a new wave of application of recommendation systems in the e-commerce field. Foreign agricultural e-commerce giants, such as LocalHarvest and Hello Fresh, have also deployed their own recommendation

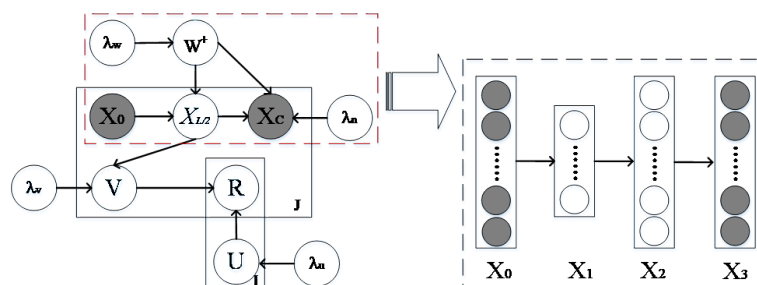
<sup>†</sup>Corresponding author.Email address: [306261663@qq.com](mailto:306261663@qq.com)

systems one after another and achieved considerable economic growth. In China, although the research on recommendation system in China is relatively late, there are also several representative e-commerce platforms that deploy recommendation systems, such as Taobao, JD.COM and Alibaba. Recently, based on the concept of ‘precise poverty alleviation’, e-commerce of agricultural products is gradually entering people’s vision as a new economic income-generating point. There are also been e-commerce giants of agricultural products, such as JD.COM Fresh, Fruit Day, No. 1 Fresh and HQW.COM, in China. All these e-commerce platforms provide users with good shopping experience by constructing multiple recommendation models according to their own business characteristics. Furthermore, they have driven the wave of online sales of agricultural products. This shows the importance of recommendation system in the field of agricultural products e-commerce. A good recommendation system can not only improve users’ loyalty, but also create a lot of economic growth.

## 2 Related Work

### 2.1 Collaborative deep learning

Collaborative topic regression (CTR) is the first to use LDA and probability matrix factorisation (PMF) to combine item content information and scoring matrix for collaborative training, and it has achieved good results. With the development of deep learning, Wang HAO et al. proposed a collaborative deep learning (CDL) model based on deep learning. To solve the problem that the sparse text information of CTR is inversely proportional to the validity, the model enables the Stacked Denoising Auto-Encoder (SDAE) to better extract the hidden features of the content information of items, and the hidden features are used to constrain the hidden features  $V$  of items decomposed by the PMF, which alleviates the cold start problem and further improves the recommendation effect. The model is shown in Figure 1.



**Fig. 1** Collaborative deep learning model.

As shown in the figure, on the right is the naive Bayesian SDAE proposed by the author.  $W^+$  represents the set of offset vectors and weight matrices of each layer in SDAE and  $L$  represents the number of layers of SDAE.  $X_0$  is a set of vector representing  $N$  item content information. When the noise is added, it becomes  $X_1$ . The role of SDAE is to restore  $X_1$  by making  $X_C$  approach  $X_0$  through self-supervised training, thus obtaining the product of the middle hidden layer  $X_{L/2}$ , i.e. the potential features of the content information of the item. The content information vector of the item adopts the method of bag-of-words. By constructing a dictionary containing  $S$  keywords, the content information of each item can be converted into vectors. Excluding the SDAE part on the right, the rest is the PMF part.  $M$  represents the number of users in the scoring matrix  $R$ ,  $N$  represents the number of items in  $R$ ,  $U$  represents the potential feature vector of users and  $V$  represents the potential feature vector of items. As can be seen from the figure as a whole, CDL is essentially the fusion of SDAE and PMF models, and  $X_{L/2}$  plays the role of a bridge. It connects SDAE and PMF models, thus combining the scoring matrix and the content information of items to jointly represent the user hidden vector  $V$ . Therefore, a convolutional matrix factorisation (ConvMF) with better effect has emerged.

## 2.2 Convolutional matrix factorisation

SDAE in CDL uses bag-of-words to represent the vectors of the text. Although it can automatically extract the potential features of the text, this method considers only the frequency of words, ignores the context of words, and cannot well represent the description information of items. To solve this problem, ConvMF uses CNN to process and extract potential features in the item description text. Its model is shown in Figure 2.

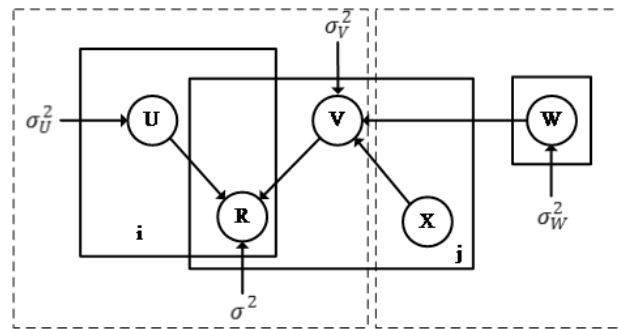


Fig. 2 Convolutional matrix factorisation model.

As shown in the figure,  $R \in R^{N \times M}$  is observation value scores,  $N$  represents the number of users in the scoring matrix  $R$ ,  $M$  represents the number of items in  $R$ ,  $U \in R^{k \times N}$  represents the potential feature vector of users,  $V \in R^{k \times M}$  represents the potential feature vector of items and  $k$  represents the potential feature. The right part of the model is the CNN model, and the left part is the PMF model.

## 3 Data Crawling

According to the actual business logic of the e-commerce websites crawled and the types of data to be crawled, a web crawler is written in a custom way, thus completing the data crawling work. The data come from large agricultural products e-commerce websites. Generally speaking, it can be divided into three parts: web access, web page analysis and data storage. The overall structure diagram is shown in Figure 3.

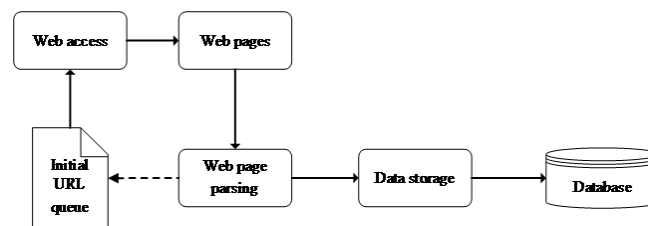


Fig. 3 Crawler structure diagrams.

### 3.1 Web access

To better complete the task of data crawling, Web Collector framework is used to build crawlers. Web Collector is an integration framework whose kernel is written in Java. Due to its feature that it does not need to configure and provide portable API interfaces, it can define crawlers that meet its own task requirements with a small amount of code, thus carrying out online data crawling. Using the initial URL queue to be crawled as the starting URL, the user-defined web crawler is started to be executed, and the page is analysed according to the

stored starting URL, thus generating the URL to be analysed.

### 3.2 Web page parsing

HTML document is a kind of text document written with unified standards. Label pairs represent various information and information display methods in web pages. Browsers can display web pages by analysing these label pairs. In general, the positions of various label pairs of HTML documents are fixed, and the HTML documents can be read and filtered by line. By writing corresponding filtering rules (such as regular expressions), these features of HTML can be utilised to obtain data.

### 3.3 Web page content storage

After parsing the data from the web page successfully, the next step is to store the data. Generally, when storing content, one can choose either relational database represented by MySQL or non-relational database represented by MongoDB. The advantage of non-relational database is that it can directly store the whole data without considering the influence of fields. Using MySQL to store data, setting the form of fields and data types in advance would take up more time and storage space, but it would be more convenient to extract or further process the data, thus reducing some preprocessing work in disguised form.

### 3.4 Data preprocessing

This article mainly uses three fields of user comment, commodity description and user score to carry out model research and experimental proof, but the initial data directly crawled from the web page are not standardised, so it is necessary to process the crawled original data to generate a training data set for later use. The specific processing steps are as follows:

1. Filter system default records
2. Clear special characters
3. Clear duplicate content. However, due to some language habits of users, some sentences, such as the common ‘say important things three times’, are repeated, which undoubtedly reduces the efficiency of opinion word retrieval.
4. Text error correction. As user comments belong to the space for users to play freely, it is inevitable that there will be some typos. However, when mining viewpoints and words, typos have a great influence, so it is necessary to correct these typos in advance.
5. Set the comment threshold. If a user has only one comment, it is difficult to obtain user characteristics through this comment, so to better complete the experiment only user records containing at least five user comments are kept.

## 4 Model

ConvMF model uses CNN to extract hidden features from item description information, which is greatly improved in comparison with the feature extraction model based on bag-of-words representation. However, the model has also three shortcomings. First, CNN’s local field of view principle can capture only short-distance context information, which would result in serious loss in context information between long-distance words. Second, the traditional static word embedding method does not consider the situation that semantics will vary with different contexts. Third, only item description information and scoring matrix are used for modelling, and user comment information is not considered. Therefore, based on ConvMF, an improved collaborative filtering model is proposed by using pretraining model BERT, bidirectional long–short memory network Bi-LSTM and user comment information.

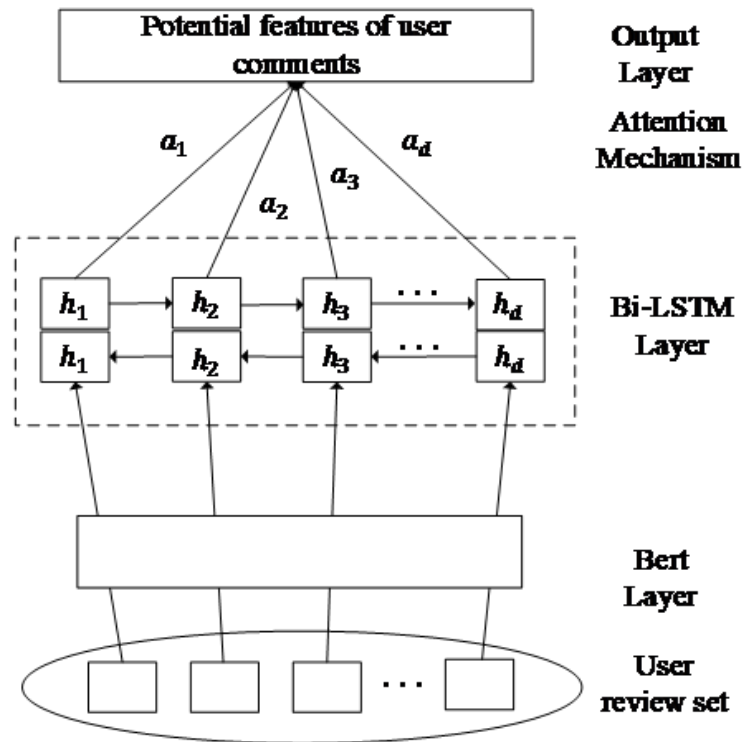


Fig. 4 User comment text processing.

#### 4.1 Handling of user comments and item descriptions

In the previous ConvMF model, the traditional Word2vec was used to construct word vectors, but the word vectors constructed in this way were static. Even in different contexts, the same word has only one coded expression, thus ignoring the situation of polysemy. For example, ‘apple’ can be either ‘fruit’ or ‘mobile phone’. It is obviously unreasonable to use the same static word vector to express it. In addition, although ConvMF uses CNN to extract certain context information from text, due to CNN’s own convolution calculation characteristics, the model cannot extract context information between long-distance words, i.e. long-term dependence. To solve this problem, the current common method is to replace the traditional Word2vec with BERT language model. As a powerful pretraining language model, BERT can construct word vectors with rich contextual semantics, which not only solves the problem of polysemy but also directly obtains sentence vectors. A user often comments on multiple items, but there is only one item description, so the processing of user comments is more complicated. Here, taking the processing of user comments as an example, the module is shown in Figure 4.

##### 1. Bert layer

For the comments generated by the user  $i$ ,  $d$  of them are selected to form the user’s comment set  $Y_i$ , i.e.  $Y_i = \{Y_{i1}, Y_{i2}, \dots, Y_{id}\}$ . The user’s comment set is processed by Bert to obtain the vector  $B \in R^{d \times c}$  and  $B_i = \{B_{i1}, B_{i2}, \dots, B_{id}\}$  of the user’s comment set. It should be noted that if the user generates more than  $d$  historical comments, the first  $d$  of the user comments will be selected to form a user comment set. If there are fewer than  $d$  comments generated,  $Y_i$  needs to be Bert processed and then several zero vectors need to be filled into  $B_i$ .

##### 2. Bi-LSTM layer

The sentence-level vector generated by each user comment after Bert layer processing contains rich contextual semantics, so it does not need to be processed by CNN like ConvMF. Bi-LSTM is used here to extract higher-order features from the entire review set. Specifically, each comment of the user reflects a certain user preference. To obtain a potential feature vector of the user, it is necessary to synthesise all the comments of the user. The forward LSTM processes the data from  $B_{i1}$  to  $B_{id}$ , while the backward LSTM processes the data

from  $B_{id}$  to  $B_{i1}$ , splicing the hidden layer states of the two LSTMs together to obtain the overall hidden layer state  $h_d \in R^{2I}$  corresponding to  $B_{id}$ , and further obtaining the Bi-LSTM hidden state  $H_i \in R^{d \times 2I}$  of each user comment set.  $I$  represents the number of hidden units of each LSTM:

$$h_d = [\bar{h}_d \oplus \tilde{h}_d]$$

$$H_i = [h_1, h_2, \dots, h_d]$$

Although each user comment can show certain user preferences, each comment obviously contributes to the potential vector of users to different degrees. Here, the hidden layer state  $h_d$  of each user comment is weighted by the attention mechanism, thus obtaining the overall expression  $A_i \in R^{d \times 2I}$  of the user  $i$  comment set:

$$a = \text{softmax}(w_1 \times \tanh(w_2 \times H_i^T))$$

$$A_i = aH_i$$

where  $w_1 \in R^{1 \times t}$ ,  $w_2 \in R^{t \times 2I}$ , and  $t$  are the super parameters of the attention layer, and the function softmax is responsible for normalising the attention weight.

### 3. Output layer

To complete our subsequent recommendation tasks, the function of the output layer here is also to map the overall expression  $A_i$  of the user comment set to the  $k$ -dimensional potential model space of users and items, thus obtaining the potential feature vector  $D_i \in R^{1 \times t}$  of user comments:

$$D_i = w_3 A_i + b_3$$

Finally, through the processing of the above steps, to facilitate the representation, all parameters appearing in the whole model are represented as  $W_D$ , so that the whole process can be regarded as a function with the original user comment text  $Y_i$  as the input and the potential feature vector of the user comment as the output, as shown in the following equation:

$$D_i = \text{bIstm}(W_D, Y_i)$$

In contrast, the feature extraction of item description text is much simpler. After using Bert to process the description text  $X_j$  of the item to obtain the vector expression of the item, the full connection layer is directly input to obtain the potential feature vector of the item. For convenience of expression, we also express the weight matrix and offset items appearing in the process as  $W_S$ , so that the potential feature vector of the item text can be expressed as:

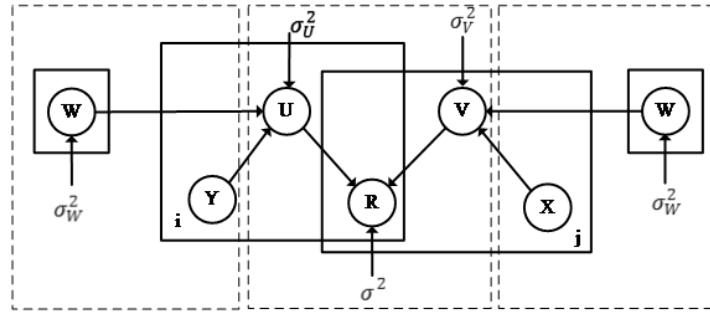
$$S_j = \text{bert}(W_S, X_j)$$

Among them,  $X_j$  represents the content description text of the item  $i$ .

## 4.2 Recommendation model based on user comments and item description

ConvMF model uses CNN to obtain better item features and successfully integrates deep learning model into collaborative filtering algorithm through PMF model. Although good recommendation effect has been achieved, it considers neither the polysemy, nor the improvement of user features to recommendation model. Therefore, to improve these problems, a recommendation model based on user comments and item description is proposed in this chapter. The model introduces both the user's comment information and the item description information, uses the feature extraction model constructed in Section 3.1 to obtain the potential features of the user and the item, respectively and finally uses these features and the potential features decomposed by PMF to combine into the final potential features of the user and the item. The specific model diagram is shown in Figure 5.

The meaning of each parameter in the figure is basically the same as that in ConvMF, and  $Y$  represents the introduced user comment text. The middle part of the model is PMF part, and the left and right sides are feature



**Fig. 5** A recommendation model based on user comments and item descriptions.

extraction part in 4.3. 2. These two parts are connected by acting on the final item hidden feature V and user feature U together. The generation process of the model is as follows:

1. For the feature extraction part, to prevent confusion, the weight and offset parameters in this part are collectively referred to as  $W$ , then each parameter in  $W$  has  $W \sim N(O, \sigma_w^2 I)$ .
2. For each user  $i$ :
  - a. Potential feature vector of user is  $\theta_i \sim N(O, \sigma_o^2 I)$  generated by PMF.
  - b. The user potential feature vector generated by the user comments is  $D_i = \text{blstm}(W_D, Y_i)$ .
  - c. Final user potential feature vector is  $u_i = \text{blstm}(W_D, Y_i) + \theta_i$ .
3. For each item  $j$ , there are:
  - a. Potential feature vector of item is  $\varepsilon_j \sim N(O, \sigma_v^2 I)$  generated by PMF.
  - b. The item description that generates the item potential feature vectors is  $S_j = \text{bert}(W_S, X_j)$ .
  - c. Final item potential feature vector is  $v_j = \text{bert}(W_S, X_j) + \varepsilon_j$ .
4. The prediction score  $r_{ij} \sim N(u_i^T v_j, \sigma^2)^{I_{ij}}$  of user  $i$  on item  $j$ , where  $I_{ij}$  is an indication function. When user  $i$  scored item  $j$ , its value is 1, and in other cases its value is 0.

The calculation of the model is similar to that of ConvMF. Since feature extraction is also carried out from user comments, the update rule of  $u_i$  and  $v_j$  becomes:

$$u_i \leftarrow (VI_i V^T + \lambda_U I_K)^{-1} (VR_i + \lambda_U \text{blstm}(W_D, Y_i))$$

$$v_j \leftarrow (UI_j U^T + \lambda_V I_K)^{-1} (UR_j + \lambda_V \text{bert}(W_S, X_j))$$

When  $U$  and  $V$  remain unchanged for the time being, the optimisation rule for  $W_D$  and  $W_S$  are updated to:

$$\varepsilon(W_S) = \frac{\lambda_V}{2} \sum_I ||V_j - \text{bert}(W_S, X_j)||^2 + \frac{\lambda W_S}{2} \sum_K ||W_k||_2 + \text{constant}$$

$$\varepsilon(W_D) = \frac{\lambda_U}{2} \sum_I ||u_i - \text{blstm}(W_D, Y_i)||^2 + \frac{\lambda W_D}{2} \sum_K ||W_k||_2 + \text{constant}$$

User  $i$ 's prediction of item  $j$ 's score is changed to:

$$r_{ij} \approx N(u_i^T v_j | \sigma^2) = (\text{blstm}(W_D, Y_i) + \theta_i)^T (\text{bert}(W_S, X_j) + \varepsilon_j)$$

## 5 Experiment

### 5.1 Data set

The data set includes 55,231 scores and comments generated by 13,239 users on 1,246 commodities. The composition of the data set is given in Table 1.



**Table 1** Data set composition

Fields	Description
UserID	User id
ItemID	Commodity id
Rating	User's rating of commodities
Review	User comment text
ItemText	Item content text
Timestap	Time stamp

**Table 2** Experimental environment settings

Environment	Description
Operating system	Centos7
Experimental language	Python 3.5.6
Deep Learning Framework	Tensorflow_gpu-1.3.0
GPU	NVIDIA Tesla
CUDA	8
cuDNN	6

## 5.2 Experimental environment and parameters

The experimental environment is shown in Table 2.

The main parameters set in the experiment are as follows: the BERT version of the pretraining model used is Chinese\_L-12\_H-768\_A-12, which is officially pretrained and open source by Google. All parameters in this section use their default parameters except the max\_seq\_length (MSL) parameter. The MSL parameter represents the text length of the input model. Text beyond this length will be truncated. The truncation length of each commodity description text will cover 90% of the commodity description, as will the truncation of each user comment. The data set is randomly divided into three parts according to the proportion of 0.8, 0.1 and 0.1, which are used as training set, verification set and test set, respectively. In addition, the number of user comments contained in each user comment set is set to 5. In the model, the number of hidden units l of LSTM is set to 200, and the dimension t of attention weight vector is set to 400. To prevent overfitting, dropout is added and set to 0.2. Some parameter settings of PMF are the same as those described by ConvMF.

## 5.3 Evaluation criteria

According to the different application fields and objects of concern, the evaluation indices of the recommendation model are also slightly different. MSE is widely used in scoring prediction tasks, so MSE is used as the evaluation standard of the experiment. The equation is as follows:

$$MSE = \frac{1}{N} \sum_{(i,j)} (R_{ij} - r_{ij})^2$$

Among them,  $R_{ij}$  is the observation value of the item j scored on user i,  $r_{ij}$  is the prediction score and  $N$  is the right number in the whole training set (user score). The higher the prediction accuracy of a model, the lower the MSE value.

## 5.4 Experimental results and analysis

The model proposed in this article is based on PMF and improved by introducing additional data sources and deep learning technology, so we use PMF, CDL and ConvMF to carry out comparative experiments under this data set. The experimental results are shown in Table 3.

According to the advantages and disadvantages of the above three models, the proposed improved model



**Table 3** Experimental results

Model	MSE
PMF	1.473
CDL	1.324
ConvMF	1.142
Our	1.031

uses BERT to represent word vectors, which not only solves the problem of polysemy of a word, but also retains rich contextual relationships. Therefore, it obtains more complete feature expression from item description. The model also introduces user comments. BERT is used to obtain rich user features, and Bi-LSTM is used to fully explore the relationship between user comments, which can not only effectively model user features, but also further alleviate the sparse problem of scoring data. Therefore, the model proposed in this article is superior to CDL and ConvMF at the same time, and the MSE value obtained by training is 1.031, which is 9.7% lower than ConvMF.

**Acknowledgement:** This research is supported by Key talent projects in Gansu Province (2020RCXM106) and Dual first-class and characteristic development guidance Special Fund project.

## References

- [1] Wang C, Blei D M. CTRing for recommending scientific items[C]. ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, Ca, Usa, August. DBLP, 2011:448-456.
- [2] Kim D, Park C, Oh J, et al. Convolutional matrix factorization for document context-aware recommendation[C]/Proceedings of the 10th ACM Conference on Recommender Systems. ACM, 2016: 233-240.
- [3] Lei Zheng, Vahid Noroozi, and Philip S Yu. 2017. Joint deep modeling of users and items using reviews for recommendation. In WSDM. ACM, 425-434.
- [4] Xiang Liang. Practice of Recommendation System [M]. Beijing: People's Posts and Telecommunications Publishing House, 2012: 103-104.
- [5] He Shunmin, Cao Wenquan. Ethical Dilemma and Regulation of Personalized Recommendation Algorithm on Online Shopping Platform – A Case Study of E-commerce “Taobao” [J]. City Journal, 2019, 40 (03): 1-6.
- [6] Lu Qibei. Research on Situated Recommendation Model for E-commerce Platform Customers' Continuous Purchase [D]. Zhejiang Industrial and Commercial University, 2016.
- [7] Hu Yi. Research on E-commerce Personalized Information Recommendation Service Model Based on Big Data [D]. Jilin University, 2015.
- [8] Xu Hailing, Wu Xiao, Li Xiaodong, Yan Baoping. Comparative Study on Internet Recommendation Systems [J]. Acta Software Sinica, 2009, 20 (02): 350-362.
- [9] Yu Li, Liu Lu. Research on personalized recommendation of e-commerce [J]. Computer Integrated Manufacturing System, 2004 (10): 1306-1313.
- [10] Deng Ailin. Research on Key Technologies of E-commerce Recommendation System [D]. Fudan University, 2003.
- [11] Ouyang Jinju, Zhu Weiping. Research on the Influence of Recommendation System and Customer Loyalty in E-commerce Websites. Modern Commerce, 2016 (9): 46-48.
- [12] Zhu Yuxiao, Lv Linyuan. Summary of recommended system evaluation indexes [J]. Journal of University of Electronic Science and Technology, 2012, 41 (2): 163-175.
- [13] Wang Guoxia, Liu Heping. Summary of personalized recommendation system [J]. Computer Engineering and Application, 2012, 48 (7): 66-76.
- [14] McAuley J, Leskovec J. Hidden factors and hidden topics: understanding rating dimensions with review text[C]. ACM Conference on Recommender Systems. ACM, 2013:165-172.
- [15] Balabanovic M, Shoham Y. Fab: content-based collaborative recommendation [J]. Comm ACM, 1997,40(3): 66-72.
- [16] Mooney R J, Roy L. Content-based book recommending using learning for text categorization//Proceedings of the 5th ACM Conference on Digital libraries. San Antonio, USA,2000:195-204.
- [17] Alex Beutel, Paul Covington, Sagar Jain, Can Xu, Jia Li, Vince Gatto, and Ed H Chi.2018. Latent Cross: Making Use of Context in Recurrent Recommender Systems. In Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining. ACM,46-54.

- [18] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. 2016. Wide & deep learning for recommender systems. In Proceedings of the 1st Workshop on Deep Learning for Recommender Systems. ACM, 7-10.
- [19] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep Neural Networks for YouTube Recommendations. In ACM Conference on Recommender Systems. 191-198.
- [20] Shuai Zhang, Lina Yao, and Aixin Sun. 2017. Deep Learning based Recommender System: A Survey and New Perspectives. (2017).