

An Effective Recommendation Algorithm Based on Multi-Source Information

Lina Tang*, Keqi Wang

College of Mechanical and Electrical Engineering, Northeast Forestry University, Harbin City, Heilongjiang, China

* Corresponding author. email: li_na_tang@126.com

Manuscript submitted November 12, 2018; accepted February 26, 2019.

doi: 10.17706/jsw.14.3.107-115

Abstract: This paper proposes an effective recommendation algorithm based on multi-source information, which employs the user feature information and image feature information to handle the problems in recommender system, such as data sparsity, cold start user problem and cold start item problem. The proposed algorithm is as follow. Firstly, this paper presents a denoising auto-encoder to handle the problem of data sparsity and cold start user problem. It can learn the hidden features with nonlinearity of user and item. In addition, the paper proposes collaborative filtering algorithm based on multi-features of items. This approach employs the convolutional neural network to extract features of the image. Then combine the features of the image and the activities of the users to solve the problems of data sparsity and cold start item problem. The proposed method mentioned above is tested with dataset called MovieLens. The results of the experiment show that the proposed method has competitive performance.

Key words: Recommendation algorithm, denoising auto-encoder, convolutional neural network, data sparsity, cold start

1. Introduction

A lot of information is produced in our daily life because of the increasing development of Internet. We need to retrieve much information in order to make a better decision. But it is difficulty to handle this problem. Therefore, the recommender system is of great importance [13-15]. And many researchers focus on this region. In reference [1], the author proposed a collaborative filtering algorithm under consideration of user activities. This approach has good performance and is widely used in industry. Some researchers put forward that collaborative filtering algorithm based on neighboring [2]. This method used different weights between users and items in neighboring. Researchers modified the similarity between user and item to get better performance [3]. More new ideas are put forward in recommender system with the development of deep learning [4]. Many neural networks are employed in this area, such as sparse auto-encoder and denoising auto-encoder [5]. And certain neural networks are used to learn the hidden features between users and items [6], [7].

However, some traditional methods [1]-[3] are only based on user activities. There are some problems in practical application, such as data sparsity, cold start user problem and cold start item problem. Data sparsity means that data of user activities is sparse. Since some users are rarely check the items. Cold start means that certain new users or items, which join the recommender system at the first time, are hard to recommend because of the lack of related information. The above mentioned problems are surely to affect

the performance of the recommender system. The approaches, which related to convolutional neural network has better performance in a way. But they are not good enough yet.

In order to solve these problems, this paper proposes an effective recommendation algorithm based on multi-source information. The proposed algorithm mainly has two contributions. Firstly, the paper presents a denoising auto-encoder to handle the problem of data sparsity and cold start user problem. It can learn the hidden features with nonlinearity of user and item. In addition, the paper proposes collaborative filtering algorithm based on multi-features of the item. This approach employs the convolutional neural network to extract the feature of the image. Then combine the features of the image and the activities of the users to solve the problems of data sparsity and cold start item problem. According to the experimental results, the proposed method has competitive performance.

The rest of this paper is organized as follow. Section 2 will show the details of the presented method. In Section 3, we will show the experimental results and the analyze them. In Section 4, conclusions will be draw.

2. Proposed Method

2.1. Denoising Auto-Encoder

Current methods are not good for handling the problems of data sparsity and cold start user problem. Therefore, this paper presents a denoising auto-encoder. We can train the module through making noise [8] in the data of user activities and considering the user information. Moreover, it can learn the hidden relationship between user and item. The details of the module are described as follow. The module consists of three layers, which are input layer, hidden layer and output layer, shown in Fig 1.

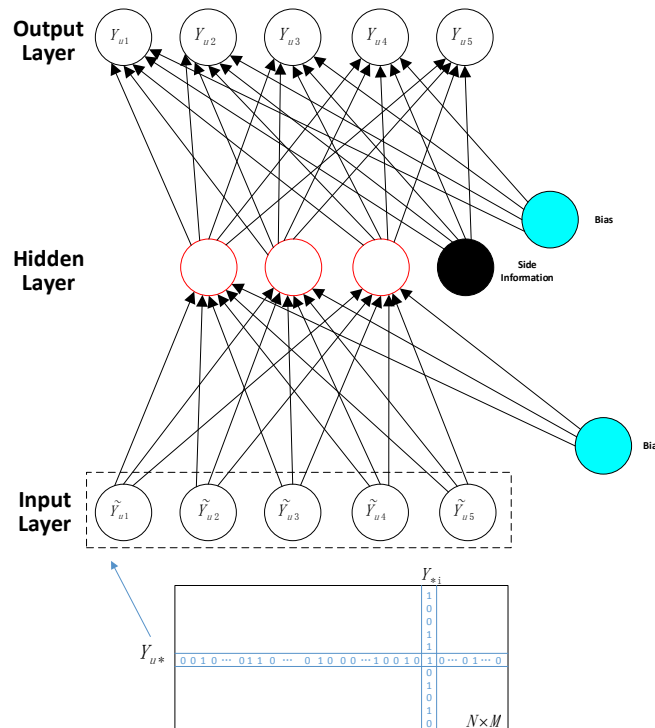


Fig. 1. construction of denoising auto-encoder.

There are M nodes in the input layer, which means M items. We make noise [8] in the user score vector Y with the possibility q . And the noising vector denotes \tilde{Y} , which is the information input in the input layer.

There are k nodes in the hidden layer, which fully connects to the nodes in the input layer. W means weight vector between input layer and hidden layer. X is the feature of the user. We combine it to the hidden layer in order to learn user feature. There are M nodes in the output layer, which represents the reconstruction value \tilde{Y} and are fully connected with the hidden layer.

When encoding, we map \tilde{Y} to hidden feature z_u , and the method is as follow:

$$z_u = h(W\tilde{Y}_u + b) \quad (1)$$

where $h(\cdot)$ represents active function, and b stands for the offset.

When decoding, we map hidden feature to original input space. \tilde{Y} is the estimation value, which can be expressed as equation (2).

$$\hat{Y}_{ui} = f(W'_i\{z_u; X_u\} + b'_i) = f(W'_i\{h(WY_u + b); X_u\} + b'_i) \quad (2)$$

X means the feature of user. We can get it from the method called TF-IDF. The method tells us that the more times that the feature appears in one user, the more important the feature is for that user. Moreover, the more times that feature appears in different users, the less contribution it can make. TF means the frequency the feature appears in the user. And it can be gotten from the equation (3)

$$TF_{ti} = \frac{f_{ti}}{\sum_{k \in Y_u^+} f_{ki}} \quad (3)$$

where f_{ti} means the number of features that appears in user u , and k means the total features that appears in the user u .

IDF means the number of feature t that appears in the total users, which can be expressed as equation (4).

$$IDF_t = \log\left(\frac{1}{1 + \sum_{j \in U} f_{tj}}\right) \quad (4)$$

So, the feature of user can be represented as equation (5).

$$X_{tu} = TF_{ti} * IDF_t = \frac{f_{ti}}{\sum_{k \in Y_u^+} f_{ki}} * \log\left(\frac{1}{1 + \sum_{j \in U} f_{tj}}\right) \quad (5)$$

The above description is the introduction of the denoising auto-encoder, which consists of three layers and two procedures---encode and decode. And the following is the introduction of target function.

The target function of the module can be expressed as equation (6).

$$\sum_{(u,i) \in Y} l(Y_{ui}, \hat{Y}_{ui}) + \lambda \phi(\theta) \quad (6)$$

where $l(\cdot)$ means the loss function, which can be expressed as equation (7), and $\phi(\theta)$ means the regularization term.

$$l(y, \hat{y}) = -y * \log\left(\frac{1}{1 + \exp(-\hat{y})}\right) - (1 - y) * \log\left(1 - \frac{1}{1 + \exp(-\hat{y})}\right) \quad (7)$$

In sum, the procedure of the proposed method is shown in figure 2.

Firstly, we make noise of the input data and initial the parameters. Secondly, we set up the max number of iteration. Thirdly, we encode and decode the information as described above. Last, the parameters are learned with stochastic gradient descent (SGD).

2.2. Collaborative Filtering algorithm Based on Multi-features of Items

In this paper, we propose a collaborative filtering algorithm based on multi-features of items to solve the problems of data sparsity and cold start item problem. We can employ the convolutional neural network to

extract the features of the image. And then combine the features of both the image and user activities to handle this problem well.

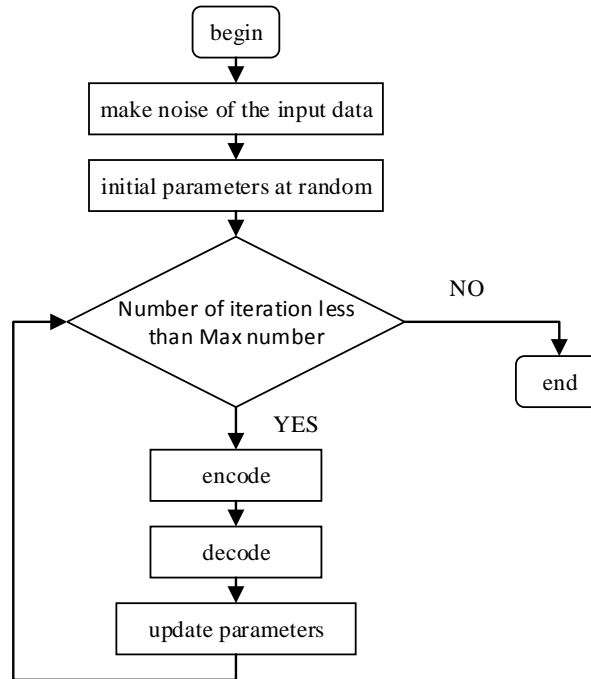


Fig. 2. procedure of the proposed method.

The proposed method consists of three steps. Firstly, we extract the features of the item through the user activities. Secondly, we combine the features of the image, extracted by CNN, and the features of user activities and get the similarity between them. Thirdly, we set up the neighbor of items and set up the list of the recommender item.

The basic construction of the convolutional neural network consists of input layer, convolutional layer, pooling layer and full connected layer, shown in Fig. 3. In this paper, we use VGG16 [9], shown in Fig. 3. It was trained with the dataset called ImageNet. The target of the training is that it can classify 1000 items. In the paper, we need to extract the features instead of classifying the objects. Therefore, what we need is the convolutional layer and pooling layer, which mark in Fig. 3 with red dotted line.

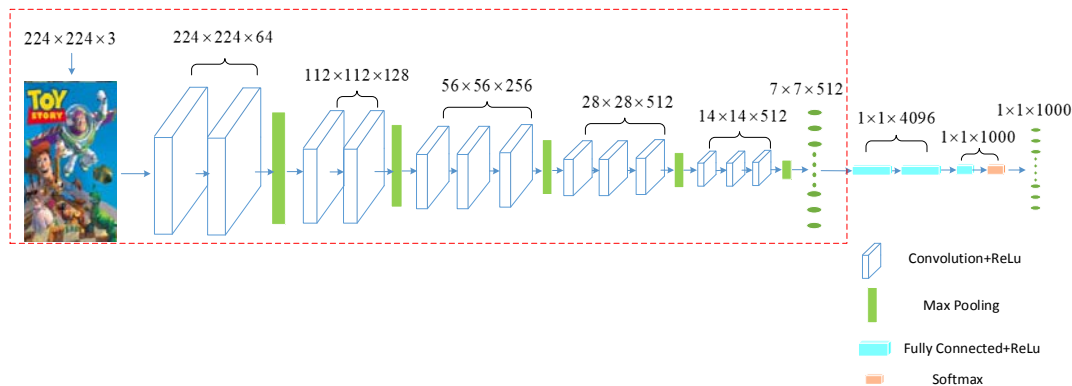


Fig. 3. construction of VGG16.

After getting the features of the image, we need to compute the similarity between them, shown in Fig. 4.

In the input layer, we convert the image into size of $224 \times 224 \times 3$. We extract the feature with size $7 \times 7 \times 512$ through the network. And then compute the similarity.

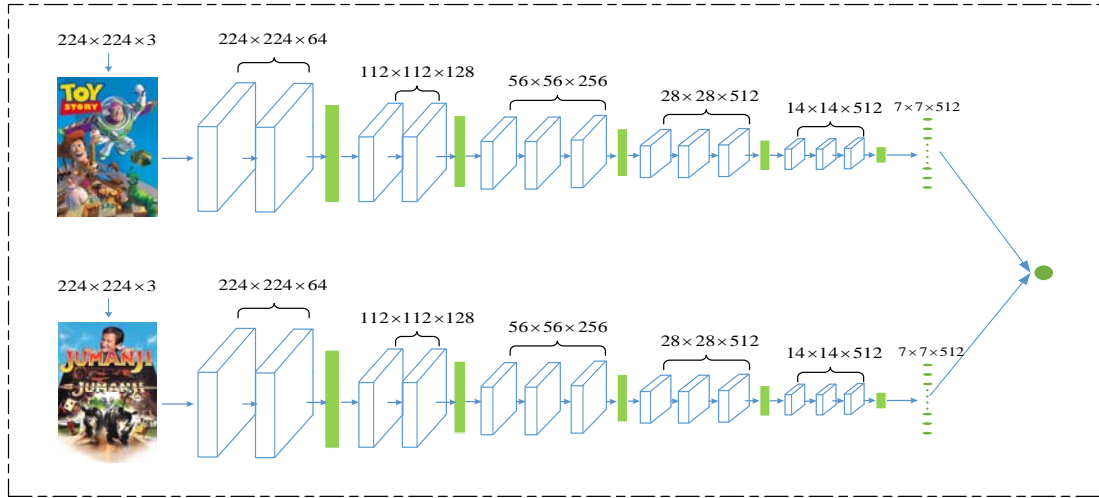


Fig. 4. Similarity.

The similarity can be gotten from equation (8).

$$\begin{aligned}
 \text{sim}(\mathbf{R}_{*i}, \mathbf{R}_{*j}) &= \frac{\left(\frac{1}{\log(1 + \text{sum}(\mathbf{R}, \text{axis} = 1))} \circ \mathbf{R}_{*i} \right) \bullet \mathbf{R}_{*j}}{\|\mathbf{R}_{*i}\|_2 \|\mathbf{R}_{*j}\|_2} \\
 &= \frac{\sum_{u=1}^N \frac{\mathbf{R}_{u,i} \mathbf{R}_{u,j}}{\log(1 + \sum_{k=1}^M \mathbf{R}_{u,k})}}{\sqrt{(\sum_{u=1}^N \mathbf{R}_{u,i}^2)(\sum_{u=1}^N \mathbf{R}_{u,j}^2)}}
 \end{aligned} \tag{8}$$

where means Hadamard product and $\text{sum}(\mathbf{R}, \text{axis}=1)$ means the vector of summation of each column in \mathbf{R} . \mathbf{R} is the score matrix.

3. Experimental Result

In this part, we implement the proposed method with the dataset called MovieLens. For example, ml-lm data set consists of three files, which deliver the information about rating, users and movies. For the typical recommendation algorithm, people always employ the information about rating. And the data in data set is sparse. The data set consists of 6040 users' rating on 3900 movies. And the total number of record is 1000209. The mean value of movie that users rate is 165. The largest number of movies rating by one user is 2314 and the least is 20. From the above analysis, we know that the data is sparse.

We implement the proposed approach and test it in the dataset MovieLense. From the experimental results, we can know that the proposed method is appropriate for handling the problem of cold start user problem and cold start item problem. And the accuracy rises. We compare their MAP (Mean Average Precision) and recall in order to show the advantages of the proposed method. MAP means the precision of the recommendation list. Recall means the percentage of items that are in the test data set. The details are shown as follow.

3.1. Denoising Auto-encoder Method

In order to show the better performance of the denoising auto-encoder, we compare the proposed method with the following methods, which are ItemRM [10], U_AutoRec [11], ItemCF [12]. The results are shown in Fig. 5 and Fig. 6. The title MAP@N means the result of MAP in Top-N recommendation. The performance is similar with different N. The presented method has best performance among them. In the proposed method, MAP@10 is 40% better than the second one. And Recall@10 is 40% better than the second one.

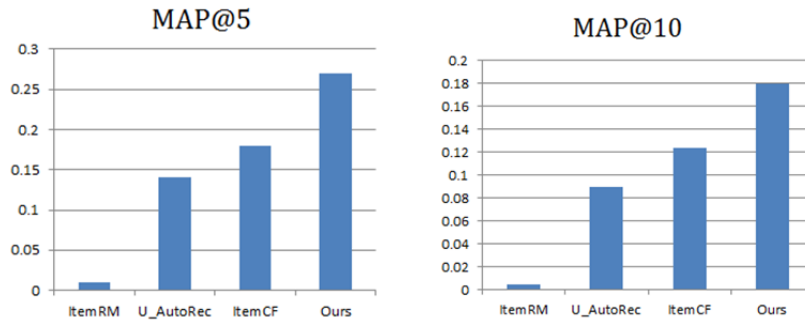


Fig. 5. MAP in denoising auto-encoder

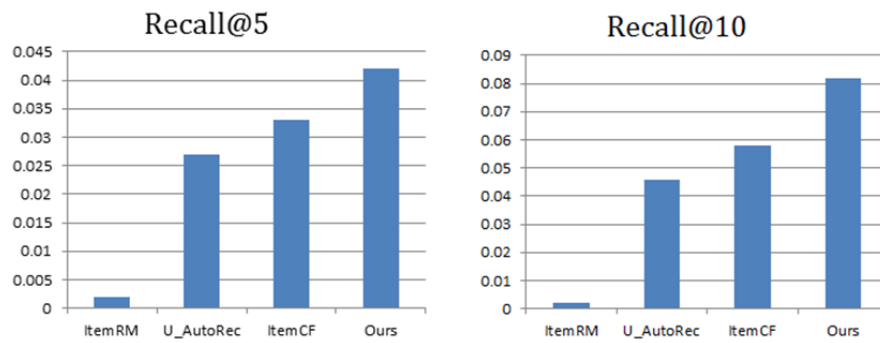


Fig. 6. Recall in denoising auto-encoder.

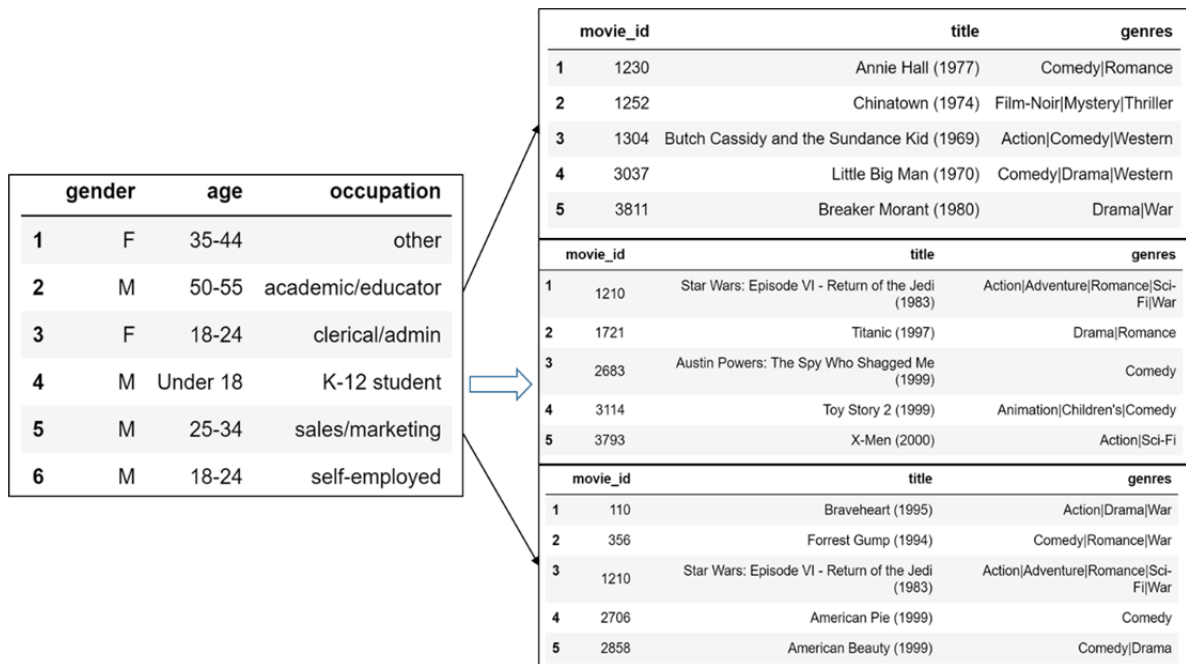


Fig. 7. test of cold start user problem.

In order to test the performance of the cold start user problem, we set up a new user. The result is shown in Fig. 7, which the user information is shown in the left and the recommendation list is on the right. Take No.4 user for example. It is proper for student to enjoy Star War, Titanic, Toy Story etc. So we can know that the precision of the list can be promoted by the user information.

3.2. Collaborative Filtering Algorithm Based on Multi-Features of the Item

In this part, we get the imdbID from the data set so that we can get the poster from the Internet. We extract the feature of the poster and count the similarity between them. The result is shown in Fig. 8. The picture with red box in the left is the original poster. The others are the similar poster recommended by the algorithm. From the result, we know that they have similar features. The first line is the Toy Story. The recommended movies are cartoon. And it is reasonable.



Fig. 8. Similarity between images

Fig. 9 is the result of experiment. From the chart, we can know that the precision and the coverage is better than the algorithm entitled ItemCF [12], which shows that new items can be recommended to users.

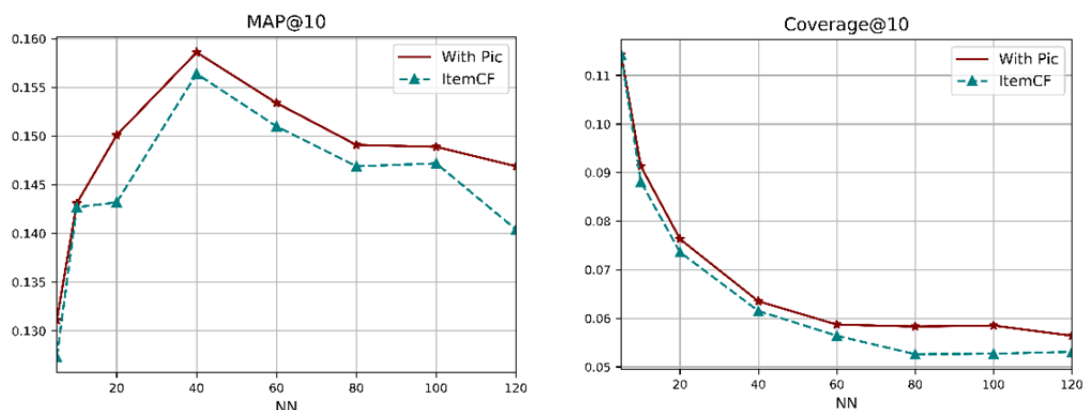


Fig. 9. MAP and Coverage in Collaborative Filtering algorithm based on multi-features of the item

4. Conclusion

This paper proposes an effective recommendation algorithm based on multi-source information. It can

solve the problems of data sparsity, cold start user problem and cold start item problem. Firstly, we employ the denoising auto-encoder, which can learn the relationship between user and item from user information. Secondly, we utilize collaborative filtering algorithm based on multi-features of items to handle the problems of data sparsity and cold start item problem. We validate the performance of the proposed method in the data set called MovieLense, which shows that the presented approach has better performance.

Acknowledgment

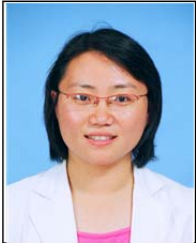
This work is supported by the project called special funds for basic scientific research in central universities (No.2572015BB23). The title is research on decision analysis under the assumption of bounded and rational condition.

References

- [1] Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender system: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6), 734-749
- [2] Breese, J. S., Heckerman, D., & Kadie, C. (2013). Empirical analysis of predictive algorithms for collaborative filtering. *Uncertainty in Artificial Intelligence*, 98(7), 43-52
- [3] Yu, K., Xu, X., Ester, M. *et al.* (2003). Feature weighting and instance selection for collaborative filtering: An information-theoretic approach. *Knowledge and Information Systems*, 5(2), 201-224.
- [4] Singhal, A., Sinha, P., & Pant, R. (2017). Use of deep learning in modern recommendation system: A summary of recent works. *International Journal of Computer Applications*, 180(7), 17-22
- [5] Scholkopf, B., Platt, J., & Hofmann, T. (2006). Greedy layer-wise training of deep networks. *Proceedings of the International Conference on Neural Information Processing Systems* (pp. 153-160).
- [6] Kawale, J., & Fu, Y. (2015). Deep collaborative filtering via marginalized denoising auto-encoder. *Proceedings of the ACM International Conference on Information and Knowledge Management* (pp. 811-820).
- [7] Wu, Y., Dubois, C., Zheng, A. X., *et al.* (2016). Collaborative denoising auto-encoders for top-N recommender systems. *Proceedings of the ACM International Conference on Web Search and Data Mining* (pp. 153-162).
- [8] Vincent, P., Larochelle, H., & Lajoie, I., *et al.* (2010). Stacked denoising criterion. *Journal of Machine Learning Research*, 11(12), 3371-3408
- [9] Simonyan, K., & Zisserman, A. (2015). Very deep convolution networks for large-scale image recognition. *Proceedings of the International Conference on Learning Representations*.
- [10] Zhang, S., Yao, L., Sun, A, *et al.* (2017). Deep learning based recommender system: A survey and new perspectives.
- [11] Sedhain, S., Menon, A. K., Sanner, S., *et al.* (2015). AutoRec: Autoencoders meet collaborative filtering. *Proceedings of the International Conference on World Wide Web* (pp. 111-112).
- [12] Sarwar, B., Karypis, G., Konstan, J., *et al.* (2001). Item-based collaborative filtering recommendation algorithms. *Proceedings of the International Conference on World Wide Web* (pp. 285-295).
- [13] Yu, C. C., Tang, Q. S., Zheng, L, *et al.* (2018). A recommender system for ordering platform based on an improved collaborative filtering algorithm. *Proceedings of the 2018 International Conference on Audio, Language and Image Processing*.
- [14] Singh, R. A. V., & Mayank, S. (2017). Building an effective recommender system using machine learning based framework. *Proceedings of the 2017 International Conference on Infocom Technologies and*

Unmanned Systems (pp. 215-219).

- [15] Kunal, S., Akshay, K. S., Saurabh, D., *et al.* (2017). Recommender systems: An overview of different approaches to recommendations. *Proceedings of the 2017 International Conference on Innovations in Information, Embedded and Communication Systems* (pp. 1-4).



Lina Tang was born in Taian city, Shandong, China. Doctor Tang received both her bachelor degree and master degree in Shandong Normal University, Jinan City, Shandong, China. Her major is educational technology. And she is pursuing her doctor degree in Northeast Forestry University, Harbin City, Heilongjiang, China. And her major is agriculture information technology.

She is the teacher in Northeast Forestry University College of Information and Computer Engineering, Harbin city, Heilongjiang, China. Her research interests are data management and signal processing.



Keqi Wang is a professor in College of Mechanical and Electronical Engineering. Prof Wang received his bachelor degree in Northeast Forestry College, Heilongjiang, China. His major is automatic control. He received his master degree is Harbin College of shipbuilding engineering, Harbin, Heilongjiang, China. His major is automatic control. And he received his doctor degree in Northeast Forestry University, Harbin, Heilongjiang, China. His major is forestry engineering.

He is a professor in College of Mechanical and Electronical Engineering, Northeast Forestry University, Harbin city, Heilongjiang, China. Her research interest is automatic control. He is the administrator in the department of forestry engineering