A Stable Collaborative Filtering Algorithm for Long Tail Recommendation

Kun Zhao¹, Jiaming Pi²,³,*

¹ Information School, Yunnan University of Finance and Economics, Kunming, China
² Business School, Yunnan University of Finance and Economics, Kunming, China
³ Zhonghua Vocational College, Yunnan University of Finance and Economics, Kunming, China

*Corresponding author e-mail: 277352342@qq.com

Abstract. As a relatively successful recommendation system, the collaborative filtering recommendation system (CFRS) has been widely used in e-commerce. However, the current CFRS is mainly based on mainstream or popular products to recommending similar items for users and is less efficiency in recommend the so called "Long Tail" products to meet the individual needs of users. Based on the Item-based system filtering recommendation algorithm, this paper proposes a collaborative filtering recommendation algorithm that implements long tail recommendation by using the item rating probability matrix and item rating reliability. Compared with the traditional collaborative filtering algorithm, the experimental result based on MovieLens 1M dataset shows that the proposed algorithm can deal with the data sparsity problem better, and is better for producing recommendation for the long tail products effect, and furthermore, it shows stability to a certain extent in producing recommendations under different situations of data sparseness.

1. Introduction

The concept of The Long Tail was first proposed by Chris Anderson, editor-in-chief of Wired magazine, in the October 2004 [1], which was used to described the business and economic models of sites like Amazon and Netflix. The long tail is not a purely theoretical result, but an economic phenomenon obtained through a lot of statistical results. This phenomenon not only has influences on e-commerce, but also affects the collaborative filtering recommendation system for important application in e-commerce.

However, most of the existing collaborative filtering recommendation algorithms can only provide recommendations for mainstream goods or services. For businesses who sell less well-known products or services, the recommendation system may not better recommend the goods or services at the "Long Tail" to consumers, therefore, helping the consumers find good "Long Tail" items, achieving the Long Tail recommendation has been a problem to be solved in recommendation systems.

In this paper, a stable collaborative filtering long tail recommendation algorithm is proposed, which introduces item scoring probability matrix to alleviate the data sparsity, builds the connection between mainstream items and long tail items to achieve the long tail recommendation. Introduction of the item scoring reliability improves the reliability of the prediction scoring so that the prediction scoring is more consistent with the reality. In addition, the new algorithm has stability in some sense, able to meet personalized needs of users.
2. Literature review

Many of the "Long Tail" items have short evaluation history time and less historical score data, which has caused data sparsity of user-item scoring matrix \(^2\). Due to the problem of data sparsity, there are few common scores of the mainstream items and "Long Tail" items, further causing the deviation in computing the similarity between mainstream items and "Long Tail" items, affecting the results of the most adjacent selection, causing the failure of "Long Tail" item recommendation.

Many scholars achieve long tail recommendation through alleviating data sparsity in the current research. The main methods are:

1. Adjust the similarity computation method of the user or item. Literature \(^3\) alleviates data sparsity by reducing the similarity among users with fewer common scoring items. Literature \(^4\) introduces similarity of items as the weight for the similarity computation among users. Literature \(^5\) applies the long tail theory to balance the similarity of users. Literature \(^6\) proposes the collaborative filtering improvement algorithm based on LDA, changing the computation approach for the similarity among users.

2. Adjust selection of the most adjacent neighbour. Literature \(^7\) introduces the user score information entropy to adjust the neighbour selection. Literature \(^8\) selects the most adjacent for the user against each target item combined with the item label.

3. Fill the user-item scoring matrix. Literature \(^9\) applies the method of mean value prediction to fill the matrix. Literature \(^10\) applies the method of social relationship and condition complement. Literature \(^11\) applies the method of multi-dimensional data filling. Literature \(^12\) applies the method of bidirectional clustering and smooth filling. While in each item type cluster, Literature \(^13\) fills the sparse matrix.

4. Build the user reliability model. Literature \(^14\) introduces the successful user recommendation probability to alleviate the data sparsity. Literature \(^15\) proposes a collaborative filtering algorithm applying network enhancement. Literature \(^16\) proposes a collaborative filtering recommendation algorithm based on user’s trust.

5. Data mining or other theoretical modelling is applied. Literature \(^17\) applies the method of clustering only long-tailed items for long-tail recommendation. Literature \(^18\) applies the probability to establish its mapping relationship with the satisfaction interval for long tail recommendation. Literature \(^19\) explores users' interest in item categories to predict the recommendations. Literature \(^20\) applies fuzzy clustering and improved SFLA for collaborative filtering recommendation. Document \(^21\) introduces a method of diluting the probability of multi-label attributes. Literature \(^22\) applies the demographic clustering algorithm to solve problems of data sparsity. Literature \(^23\) proposes a long tail recommendation algorithm based on graphs.

The above research has alleviated problems of data sparsity to a certain extent, though the long tail recommendation has been achieved. But there are still problems as follows:

1. Many methods only improve the accuracy of the recommendation by solving the problem of data sparsity, thus achieving the long tail recommendation. Or use a certain method to separate the mainstream and "long tail" items, and only recommend long tail items. And in these methods, it is not considered to establish connection between mainstream items and "long tail" items to achieve the long tail recommendation while solving the problem of data sparsity.

2. On balancing recommendation of the mainstream items and the “long tail” items, the long tail item was excessively concerned, and the reliability of the scoring of the mainstream item is not considered, that is, the reliability problem of the prediction score and the stability of the algorithm are not considered. Although some research has established a user reliability model, it has not been combined with long tail recommendations.

Aiming at the above-mentioned problems, this paper introduces the item scoring probability matrix and item scoring reliability to solve the above-mentioned problems. The specific description is as follows:

1. With the user-item scoring matrix, from the perspective of the item, the scoring probability matrix of the item is established to solve the problem of data sparsity. Based on this, apply the item scoring matrix to compute the similarity between items, establish the connection between mainstream
items and long tail items to improve the accuracy of the similarity computation, in order to achieve the long tail recommendation.

(2) Introduce the item scoring reliability, when considering the long tail items, also pay attention to the influences of mainstream items on the prediction scoring computation and improve the reliability of the prediction scoring, so that there is stability to a certain extent in the algorithm.

3. Algorithm Design

3.1. Probability Matrix of Item Rating

Based on the above-mentioned discussion, this paper proposes the item scoring probability matrix to deal with the problem of data sparsity, establishes the connection between mainstream items and long tail items, so that the computation of similarity between items is more reasonable and accurate, thus achieving the long tail recommendation.

The item scoring probability reflects the degree of word-of-mouth and user acceptance in the user group. For example, the probability of a 5-score evaluation by users in the total scores, the probability of a 4-score evaluation by users in the total scores, probability of these scores represent to what extent the word-of-mouth and acceptance of users is, therefore, it is realistic to apply the scoring probability instead of the score to calculate the similarity. The details are as follows.

For a common collaborative filtering recommendation system with m users and n-item 5-score evaluations, Equation (1) can be applied to compute the scoring probability of the item.

\[
Pr_{is} = \frac{n_{is}}{\sum_{s=1}^{5} n_{is}}
\]

Where, \(Pr_{is}\) is the score probability of 5 for item \(i\), and \(Pr_{i1} + Pr_{i2} + \ldots + Pr_{i5} = 1\), \(n_{is}\) is the number of score \(s\) of users on item \(i\), and \(\sum_{s=1}^{5} n_{is}\) is the total amount of user’s score on item \(i\), and \(\sum_{s=1}^{5} n_{is} \neq 0\), i.e., the cold boot brought about by the new item is not considered.

After the score probability of all items are computed, the user-item scoring matrix \(R_{mn}\) can be converted into the item scoring probability matrix \(Pr_{n \times 5}\) (because it is a five-level score system, and if it is a 7-level score system, the item scoring probability matrix should be \(Pr_{n \times 7}\)), shown as in Table1.

| Table 1. User-Item Rating Matrix. |
|----------------------------------|
| Item_1 | \(R_{11}\) | \(\ldots\) | \(\ldots\) | \(R_{1n}\) |
| \(\ldots\) | \(\ldots\) | \(\ldots\) | \(\ldots\) | \(\ldots\) |
| \(\ldots\) | \(\ldots\) | \(\ldots\) | \(\ldots\) | \(\ldots\) |
| User_m | \(R_{m1}\) | \(\ldots\) | \(\ldots\) | \(R_{mn}\) |

Convert the user-item scoring matrix into the item scoring probability matrix using Equation (1), as shown in Table2:

| Table 2. Item Rating Probability Matrix. |
|---------------------------------|
| Number | 1 | 2 | 3 | 4 | 5 |
| Item_1 | \(Pr_{11}\) | \(Pr_{12}\) | \(Pr_{13}\) | \(Pr_{14}\) | \(Pr_{15}\) |
| \(\ldots\) | \(\ldots\) | \(\ldots\) | \(\ldots\) | \(\ldots\) | \(\ldots\) |
| \(\ldots\) | \(\ldots\) | \(\ldots\) | \(\ldots\) | \(\ldots\) | \(\ldots\) |
| Item_n | \(Pr_{n1}\) | \(Pr_{n2}\) | \(Pr_{n3}\) | \(Pr_{n4}\) | \(Pr_{n5}\) |
The user-item scoring matrix is converted into the item scoring probability matrix, which alleviates the deviation of the similarity computation caused by the matrix sparsity to a certain extent, and establishes the connection between the mainstream items and the long tail items. After it is converted into the scoring probability, even if the similarity computation can be performed to the long tail items with few scores and mainstream items with a lot of scores, which improves the accuracy of the similarity computation, thus improving the possibility that the long tail item is recommended. Therefore, the item scoring probability matrix also achieves the long tail recommendation while solving the data sparsity problem.

3.2. Item Rating Reliability

In the recommendation system, quantity of scores by users on the mainstream item is more than the quantity of scores on the long tail items, which makes users' score on the mainstream item more credible, so the impact of mainstream items should be increased when predicting score computation to improve the reliability of the predicted scores. This paper proposes item scoring reliability weights to balance the impact of mainstream items and long tail items on the prediction score computation results, so that the prediction score computation results are more reliable. The specific scoring reliability weights are described below.

In step 1, the weight of the score quantity of the item is computed using the following equation. The weight of the score of item i is computed shown as in Equation (2):

$$W_{-N_i} = \frac{N_i - N_{\text{min}}}{N_{\text{max}} - N_{\text{min}}}$$  \hspace{1cm} (2)

Where, \(W_{-N_i}\) represents of weights of the quantity of score i, and \(0 \leq W_{-N_i} \leq 1\), \(N_i\) represents quantity of scores on item i by users, \(N_{\text{max}}\) represents quantity of the item with the most scores by user, \(N_{\text{min}}\) represents quantity of the item with the least scores by user.

In step 2, apply the below equation to compute the score reliability weight of the item to be predicted, and computation of the score reliability weight of an item to be predicted, item i, is shown as in Equation (3):

$$W_{-R_i} = \frac{1}{W_{-N_i}} + W_{-N_j}$$  \hspace{1cm} (3)

Where \(W_{-R_i}\) represents the scoring reliability weight of item i, and \(W_{-N_i}\) represents the score quantity weight of item i, and \(W_{-N_j}\) represents the score quantity weight of the nearest adjacent item j of item i.

It can be concluded from Equation (3) that when the nearest adjacency item (item j) to item i is a mainstream item, its weight \(W_{-N_j}\) is large and has a more significant influence on the prediction score, improving the reliability of the prediction score. Also, weight of the score quantity of the item to be predicted is considered, to improve the foundation of the reliability of the prediction score, further improving the accuracy of the long tail item recommendation, thus better achieving better long tail recommendation.

3.3. Similarity and Prediction Calculation

According to the conclusions drawn by the researchers on the comparison of various measurement methods through experiments [24,25], the Pearson correlation similarity measure is applied as the benchmark method of user similarity measurement in this paper. The computation equation of \(\text{sim}(i,j)\) is as shown in Equation (4):

$$\text{sim}(i,j) = \frac{\sum_{s=1}^{k}(Pr_{is} - \overline{Pr})(Pr_{js} - \overline{Pr})}{\sqrt{\sum_{s=1}^{k}(Pr_{is} - \overline{Pr})^2} \cdot \sqrt{\sum_{s=1}^{k}(Pr_{js} - \overline{Pr})^2}}$$  \hspace{1cm} (4)
Where $sim(i,j)$ is the similarity of items $i$ and $j$; $Pr_i$ is the probability that item $i$ is scored as $s$; $Pr_i$ is the average probability of all scores of all items $i$; $Pr_j$ is the probability that item $j$ is scored as $s$; $Pr_j$ is the average probability of all scores for all items $j$.

The score prediction equation is shown in Equation (5).

$$P_{(u,j)} = r_j + \sum_{i \in I_u} W_{R_j} \times sim(i, j) \times (r_{(u,i)} - r_i)$$

Where $P_{(u,j)}$ represents the predicted score of item $j$, and $\bar{r}_i$ and $\bar{r}_j$ respectively represent the weighted average score obtained by multiplying the scores of item $i$ and item $j$ by the probability of the corresponding score. $I_u$ represents the set of items scored by the user $u$, $W_{R_j}$ represents the score reliability weight of item $j$ to be predicted, $r_{(u,i)}$ represents the score of user $u$ for item $i$, and $sim(i, j)$ represents the similarity of between item $j$ to be predicted and its nearest adjacency item $i$.

4. Experiments and Result Analysis

4.1. Experiment Data Set

This paper applies MovieLens’ 1M dataset as an experimental dataset, which includes 1,000,209 scores from 3,040 users for 3,952 movies. The user score is an integer between 1 and 5, and each user scores at least 20 movies with a sparsity of 0.9581.

4.2. Evaluation Metrics

The evaluation indicators used in this paper are: MAE (Mean Absolute Error, Equation (6)), RMSE (Root Mean Square Error, Equation (7)), Precision (Equation (8)) and Recall (Equation (9)) as Test indicators for algorithmic performance. The MAE test is applied to examine and evaluate the overall accuracy and stability of the algorithm. RMSE is further applied to test the overall stability of the algorithm. Precision and Recall are used to test the recommended results. Accuracy and stability of long tail recommendation is tested through combining the above-mentioned indicators.

$$MAE = \frac{1}{N} \sum_{i=1}^{N} |p_i - q_i|$$

Where $N$ is the quantity of predicted scoring items in the system, and $p_i$ and $q_i$ are the predicted score and the actual score, respectively. The smaller the value, the better the recommendation algorithm is.

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (p_i - q_i)^2}$$

Where $N$ is the number of predicted scoring items in the system, and $p_i$ and $q_i$ are the predicted score and the actual score, respectively. The smaller the value, the better performance of the recommendation algorithm is.

$$P = \frac{N_{rs}}{N_s}$$

$N_{rs}$ represents the quantity of items preferred by the user in the recommendation list, and $N_s$ represents the quantity of all recommended items. The larger the value, the better performance of the recommendation algorithm is.

$$R = \frac{N_{rs}}{N_r}$$
$N_{rs}$ represents the quantity of items preferred by the user in the recommendation list, and $N_r$ represents the quantity of all recommended items. The larger the value, the better performance of the recommendation algorithm is.

4.3. Results Analysis

Figure 1 illustrates the MAE results obtained from the traditional item-based collaborative filtering recommendation algorithm (IBCF) and the long tail recommendation algorithm (LTCF) proposed in this paper, and Figure 2 illustrates the RMSE results of the two. The MAE and RMSE values were obtained under different number of the nearest neighbor from 5 to 60 with an interval of 5 are plotted for the two algorithms in the table.

![Figure 1. MAE.](image)

From Figure 1, we can conclude that the MAE value of the LTCF is lower than that of the IBCF on the whole, indicating that the LTCF is superior to the IBCF. The MAE of the LTCF tends to be stable after the NN number is taken as 45. The difference between the maximum and minimum MAE values of the LTCF is less than the difference between the maximum and minimum MAE values of the IBCF. From Figure 2, we can also conclude the similar conclusions for the two algorithm.

Because both MAE and RMSE of the LTCF tend to be stable when the NN number is taken as 45, therefore, we checked the Precision and Recall of the two algorithms at this point, as shown in Figure 3 and 4 separately. In these two figures, the horizontal ordinate represents the number of items the algorithms would recommend to users (TopN value). The TopN value increases from 5 to 20.
It can be concluded from the Figure 3, the accuracy of both algorithms increases first and then decreases as the TopN value increases. When the TopN is 12, the accuracy of the LTCF is maximized. When the TopN is 14, the accuracy of the IBCF is maximized. It can be seen from the Figure 3 that the accuracy of the LTCF is superior to that of the IBCF, indicating that the recommendation performance of the LTCF is superior to that of the IBCF.

It can be seen from Figure 4 that the recall rate of the LTCF is slightly superior to that of the IBCF, indicating that the recommendation performance of the LTCF is superior to that of the IBCF.

From the results of our experiment as shown above, we can conclude as following:
(1) The LTCF can better achieve long tail recommendation, and its accuracy and stability are better than the IBCF.
(2) The recommendation performance of the LTCF is superior to that of the IBCF.

5. Conclusion
The LTCF proposed in this paper uses the user-item scoring probability matrix to compute the similarity, which alleviates the problem of data sparsity to some extent and establishes the connection between mainstream items and long tail items, so that the computation of similarity between items is more accurate. Based on this, the item score reliability weight is introduced, which can improve the reliability of computation results of the prediction score and produce long tail recommendation better, thus improving the recommendation performance, and its stability is superior to that of the IBCF. This research provides new research ideas and methods for solving the problem of matrix sparsity and
achieving the long tail, and improves the recommendation performance to meet the individual needs of users, so as to enable e-commerce website users to have a better recommendation experience. Therefore, this research has a certain theoretical and practical significance.

In our further research, we will strive to improve the item reliability weight computation method to further improve the accuracy of the algorithm, and it is necessary to conduct experiments by using different data set to verify the accuracy and effectiveness of the algorithm.

Acknowledgments
This work was financially supported by the National Natural Science Foundation Program of China [71462036].

References
[1] Wu Qingwei, Hong Tao, Ma Jun. A Review of Long Tail Theory[J]. Journal of Zhou Kou Normal University. 2010,(1): 124-129.
[2] Xu R, Wang S, Zheng X, et al. Distributed collaborative filtering with singular ratings for large scale recommendation[J]. Journal of Systems and Software, 2014, 95: 231-241.
[3] Huete J F, Fernández-Luna J M, de Campos L M, et al. Using past-prediction accuracy in recommender systems[J]. Information Sciences, 2012, 199: 78-92.
[4] Keunho. A new similarity function for selecting neighbors for each target item in collaborative filtering[J]. Knowledge-Based Systems, 2013, 37(1): 146–153.
[5] Zheng Suyang, Jiang Jiulei, Wang Xiaofeng. Collaborative Filtering Recommendation Algorithm Based on User Project Experience[J]. Computer Engineering, 2017, 34(8): 215-218.
[6] Liu Yi, Feng Wei, Wei Tongtong, et al. An Improved Collaborative Filtering Recommendation Algorithm[J]. Computer and Modernization. 2017, (1): 1-4, 12.
[7] Kaleli C. An entropy-based neighbor selection approach for collaborative filtering[J]. Knowledge-Based Systems, 2014, 56: 273-280.
[8] Zhang Jinglong, Huang Mengxing, Zhang Yu, et al. Collaborative Filtering Recommendation Algorithm Based on Label Optimization[J]. Journal of Computer Applications. 2018,(10).
[9] Jiang Zongli, Wang Wei, Lu Chen. Improvement of Collaborative Filtering Recommendation Algorithm Based on Mean Estimation[J]. Computer Technology and Development. 2017, (5): 1-5.
[10] Zhang Weimin, Li Rulu, Li Yongli. Collaborative filtering recommendation algorithm based on social relations and conditional complementation[J]. Journal of Jilin University (Science Edition). 2017, (5): 1244-1248.
[11] Song Meiqing. A collaborative filtering recommendation algorithm based on multi-dimensional data filling [C]/ Proc of the 2nd IEEE International Conference on Computer and Communications. 2016: 175-179.
[12] Wei Suyun, Ye Ning, Zhang Shuo, et al. Collaborative filtering recommendation algorithm based on item clustering and global similarity [C]/ Proc of the 5th International Conference on Business Intelligence and Financial Engineering. Washington DC: IEEE Computer Society, 2012: 69-72.
[13] He Ming, Sun Wang, Xiao Run, et al. A Collaborative Filtering Recommendation Algorithm Based on Fusion Clustering and User Interest Preference [J]. Computer Science. 2017,(A2): 391-396.
[14] Li Yungming, Wu Chunte, Lai Chengyang. A social recommender mechanism for e-commerce: Combining similarity, trust, and relationship [J]. Decision Support Systems, 2013, 55(3): 740-752.
[15] Li Yichen, Chen Li, Shi Chenchen, et al. Collaborative filtering algorithm enhanced by trust network [J]. Computer Application Research. 2018, (1).
[16] Li Liang, Dong Yuxin, Zhao Chunhui, et al. Collaborative Filtering Recommendation Algorithm Based on User Trust[J]. Microcomputer Systems. 2017, (5): 951-955.
[17] Feng Yuanyuan, Wang Xiaodong, Yao Yu. Recommended method for long tail items in e-commerce[J]. Computer Applications. 2015, (A2): 151-154.
[18] Pan Taotao, Zhu Wei, Wu Yitao. Collaborative filtering recommendation algorithm based on satisfaction interval[J]. Journal of Computer Applications, 2017, 34(08): 2282-2286.
[19] Dang Bo. Collaborative filtering algorithm based on score difference and user preference [J]. Computer Applications, 2016, 36(4): 1050-1053.

[20] Xu Zhihong, Tian Yu, Yan Wenjie, et al. Collaborative filtering recommendation based on fuzzy clustering and improved hybrid frog leaping [J]. Computer Application Research. 2018, (10).

[21] Chen Ying, Hou Huimin. Collaborative Filtering Recommendation Algorithm Based on Project Attribute Preference Mining [J]. Computer Applications. 2017, (A1): 262-265.

[22] HUANG Y X, CUI B, ZHANG W Y, et al. TencentRec: Realtime stream recommendation in practice [C] / / Proceedings of ACM SIGMOD International Conference on Management of Data 2015: 227-238.

[23] H Yin, B Cui, J Li, J Yao, C Chen. Challenging the long tail recommendation[J]. Proceedings of the VldbEndowment, 2012, 5 (9): 896-907

[24] Schwab I, Pohl W, Koychev I, et al. Learning to recommend from positive evidence[C]/Riecken D, Benyon D, Lieberman H. Proceedings of International Conference on Intelligent User Interfaces, New York, 2000: 241-247.

[25] Ying Xiaomin. The research on user modeling for internet personalized services[D]. National University of Defense Technology, 2003.