Personalized Resource Recommendation Based on Collaborative Filtering Algorithm

Dongmin Mo, Xue-Gang Chen *, Sheng Duan, Lu-da Wang, Qian Wu, Meiling Zhang and Lanqing Xie

College of software and communication engineering, XiangNan University, Chenzhou, 423000, China
*Corresponding Author: Xue-Gang Chen

Abstract: Digital libraries can satisfy people's demand for books and real-time information resources, but because digital libraries have huge resources, users cannot find the resources they need in time, at this time, personalized recommendation methods become an effective way to solve the problem. This paper takes collaborative filtering algorithm based on user recommendation as the research object, analyses the principle of the algorithm and the problem of sparse scoring data, an improved collaborative filtering algorithm is proposed, the algorithm sets scoring by borrowing time, and takes the sum of multiple user scoring as the scoring among users, then, according to the similarity of book content, the user's scoring of non-scored books is predicted. The proposed algorithm is applied to the digital book recommendation system to obtain better recommendation performance, compared with traditional methods, and the recommendation system based on this algorithm can get more recommendation results close to the needs of readers.

1. Introduction
With the continuous development of technology, people have more and more demands for knowledge, knowledge in the current society is more shown up in the form of libraries [1]. With the support of the Linux system and Andriod system, many developers digitalize the book resources and generate mobile libraries, with the help of mobile network tools, users can read professional books, library resources can be read on public platforms [2]. According to wikipedia's explanation, the digital library integrates the collected books with the information of the Internet, by means of information technology, by the Internet platform, users can subscribe to books in the mobile network environment [3]. At the same time, with the upgrade of the concept of universal learning, people not only need to enter the library to learn, but also need the real-time transmission of knowledge, that is, knowledge can be shared in a larger range, at the same time, it should be noted that although digital libraries[4] can store a large amount of resources, people can subscribe by mobile devices and computers, but the vast amount of information is not available purpose and lack of reasonable recommendation mechanism, which can make many users unable to quickly get the recommended resources they need.

Accurate inquiry of book resources needs the support of a large number of commodity search technologies. A large number of recommendation technologies emerged, which promoted the development of digital recommendation technology to a certain extent, among them, content-based recommendation algorithms [5] and collaborative filtering recommendation algorithms [6] were widely used. Content-based recommendation algorithm captures user's use records, it includes user's collection of resources and rating of their favorite resources, based on this, the algorithm can establish corresponding user's information description, define the type of user's resources, and generate
personalized user information files, when users send out requests of resource, the system can generate a query expression base on user's request and user's information, and calculates the matching degree of user request-user information file and request resource, obtains the resource that meets user's needs and conforms to user's characteristics, which be taken as a recommendation result, and feeds back to user terminal[7]. Among them, user-based collaborative filtering algorithm regards the object browsed by the user as the data source and the user's access history, which is suitable for occasions where data is fragmented and user types are scarce. Project heuristic-based collaborative filtering algorithm help users recommend objects or entities of their interest, in this case, the definition of entity can be described as all the objects the user likes, and an object can be mapped to multiple user personal databases.

In this paper, the user-based collaborative filtering algorithm is taken as the research object, and the implementation process of the algorithm is deeply studied, the data items of the score matrix have serious default values, an improved algorithm is proposed to compensate for the default values, the algorithm takes the reading time of books as the basis of the score, and the users with similar interests can be taken as a scoring object, then, the results of the scoring object are used for all books not scored in the user group. The algorithm is applied to the recommendation system of digital books, and the main page and each function page of the system are designed, the recommendation results before and after the introduction of the algorithms are compared and the performance of the system is analyzed.

2. User-Based Collaborative Filtering Recommendation and Its Problems

This paper chooses user-based collaborative filtering recommendation algorithm as the research object, introduces the implementation principle of the algorithm, summarizes the implementation steps of the algorithm, and studies the existing problems.

2.1. Implementation Process of Collaborative Filtering Recommendation Algorithm

User-based collaborative filtering recommendation algorithm, in the process of implementation, needs to sort out user information, and then take the way of similarity calculation to get possible results and from these results, screen out a part of the recommendation to users.

(1) Data representation

In practice, we can regard users and corresponding items as a two-dimensional table, the horizontal of the table represents the items of users’ interest, the vertical of the table can correspond to different users, that is EC×D, C represents the number of users, and D represents the maximum number of items corresponding to a certain user. From the perspective of entity, Eik represents the rating of the i reader on the entity object k, and from the perspective of literature, Eik represents the feeling of the i user after reading the material K. The evaluation should be computable, it should be expressed by logical symbols, and the range of the score should not be too large.

(2) Similarity calculation method

This step needs to select neighbors similar to the interests of the target user, they are calculated by certain algorithms, and the similarity can be calculated by the following 3 methods.

Assuming that user A and user B’s interest preferences are mapped into two vector spaces, where vector a1 corresponds to user A’s vector space and vector b1 corresponds to user B’s access space, the intersection of corresponding orientation variables can be deduced by means of cosine implicit between vectors, vector space is composed of multiple vectors, it is necessary to further expand the variables of cosine calculation into multiple vectors, and calculates this set. Assuming that user A and user B’s access records can be represented as a set of M vectors, if Tae is user A’s perception of attribute e and Tbe is user B’s perception of attribute e, then Sim (A, B) corresponding to user A and user B is interpreted as.

\[
sim(a, b) = \cos(a, b) = \frac{\sum_{e=1}^{M} T_{ae} \ast T_{be}}{\sqrt{\left(\sum_{e=1}^{M} T_{ae}^2\right)\left(\sum_{e=1}^{M} T_{be}^2\right)}} , (1 \leq e \leq M)
\]
In the process of calculating similarity, we need to avoid some irrelevant data as far as possible. Assuming that the product set of user A and user B jointly participate in rating is expressed as PAB, \( T_{ae} \) is the score of user A on vector e, \( T_{be} \) is the score of user B on vector e, \( T_a' \) and \( T_b' \) represent the average score of user A and user B. In the process of calculating the similarity between user A and user B, the variance calculation method is introduced, combines the variance calculation with cosine calculation, then the similarity \( \text{sim}(a, b) \) of user A and user B can be expressed as follows.

\[
\text{sim}(a, b) = \frac{\sum_{e \in P_{ab}} (T_{ae} - T_a')(T_{be} - T_b')}{\sqrt{\left[ \sum_{e \in P_{ab}} (T_{ae} - T_a')^2 \right] \left[ \sum_{e \in P_{ab}} (T_{be} - T_b')^2 \right]}}
\]

(2)

The cosine calculation method can be adjusted by introducing the idea of variance calculation, considering the participation data set \( P_{ab} \) of user A/user B, user A's score data set \( P_A \) and user B's score data set \( P_B \) should be larger than the joint participation data set \( P_{ab} \), then user A and user B's similarity \( \text{sim}(a, b) \) can be expressed as follows.

\[
\text{sim}(a, b) = \frac{\sum_{e \in P_{ab}} (T_{ae} - T_a')(T_{be} - T_b')}{\sqrt{\left[ \sum_{e \in P_a} (T_{ae} - T_a')^2 \right] \left[ \sum_{e \in P_b} (T_{be} - T_b')^2 \right]}}
\]

(3)

Recommending screened results

Different users recommend different results, in the actual screening process, select the results of the previous N items, the recommendation process needs to score products that are not scored. Assuming that \( S_a' \) and \( S_b' \) represent the average perception of user A and user B, B is the entity in the nearest neighbor set of user A, and \( S_{bi} \) is the perception of user B on attribute i, then user A's perception \( S_{ai} \) of attribute I that has not been scored can be expressed as.

\[
S_{ai} = S_a' + \frac{\sum_{b \in \text{next}} \text{sim}(a, b) \times (S_{bi} - S_{b'})}{\sum_{b \in \text{next}} (|\text{sim}(a, b)|)}
\]

(4)

2.2. Problems in Collaborative Filtering Recommendation Algorithm

In the traditional collaborative filtering recommendation algorithm, one of the main problems is that the data is too sparse, that is, there are many products, but the proportion of products evaluated by users is small. Under this condition, in order to find the neighbor set of the target user, it is necessary to traverse a large number of product sets, and then calculate the similarity with the target user, because the number of actual evaluations is small, it results in the inefficiency of the algorithm. At the same time, there are too many empty values in the perceptual data set, which will lead to the breakdown of the perceptual relationship between neighbor and target user.

Assuming that user A and user B have similar interest tendencies in the evaluation of some products, while user B and user C have similar interest tendencies in the evaluation of some products, the common product evaluation data of user A and user B are set as \( P_{\text{in}}^{AB} \), and the common evaluation data of user B and user C are set as \( P_{\text{in}}^{BC} \), the following states exist.

\[
\begin{cases} 
P_{\text{in}}^{AB} \cap P_{\text{in}}^{BC} = 0 \\
P_{\text{in}}^{AC} \neq 0 
\end{cases}
\]

(5)

In formula (5), from the corresponding evaluation of public goods, user A and user C may not have a common evaluation of products, but in fact, user A and user C have common preferences for products. China's book resources are growing, take Huazhong university of science and technology as an example, in 2017, the total number of books in this school has reached 3.7 million, but the number of teachers and students is 65,000, therefore, the proportion of books evaluated is very low, that is to say, the evaluation information of the whole book is sparse, the use of traditional collaborative filtering algorithm will cause problems of low efficiency and large memory consumption.
3. Improved Collaborative Filtering Recommendation Algorithm

Current collaborative filtering recommendation algorithms have matrix data default, extremely sparse scoring entity objects will result in the lack of necessary support for computing, be unable to find entity objects similar to the target users. In order to solve this problem, we can consider setting the non-scored items as a non-null value, experiments show that this method can improve the recommendation effect of the recommendation system, but there are also problems, because all the items without scoring cannot be the same value. In the process of researching this algorithm, we can adopt entity-heuristic collaborative filtering algorithm \[8, 9\] to find \(N\) entities with the highest similarity among similar entities, while most items don't have any rating, according to their similarity with the items already scored, we can put weighted method into giving an appropriate score to the items without rating, so as to alleviate adverse effects by sparse data.

3.1. Preprocessing of Book Data

The borrowing information of books comes from the library management system, the original borrowing information contains many useless information, generally, the borrowing information includes the user's ID number, the user's department, the user's type, the borrowing time, the name of the borrowed books, the address of the borrowed books, the publishing house of the books, and so on. In the process of recommendation, we only need the user's ID number, borrowing date and ISSN number of books, and other useless information can be cleared directly.

Some users may be interested in a book because of time constraints, it needs to borrow it many times, therefore, for those who borrow the same book for many times, they need to sum the borrowing time. Some users have too little information, in order to ensure the recommendation effect, the user information with less information is cleared, and the minimum borrowing value is set to 7, that is, if the borrowing value is less than 7, it means that the information cannot support recommendation, it needs to be cleared. The books in general libraries are not graded. Here the borrowing time is transformed into grading information.

Assuming that the full score of the score is 10, the maximum borrowing time is 60 days, and if the borrowing time is less than 60 days, the score can be calculated by the form of percentage, and the current score of borrowing books can be expressed as.

\[
S_{book} = \begin{cases} 
10, & t_{book} \geq 60 \\
\frac{t_{book}}{60} \times 10, & t_{book} < 60 
\end{cases}
\]  

(6)

In formula (6), \(S_{book}\) represents the recommended score for the current data to be selected, and \(t_{book}\) represents the time for borrowing books. At the same time, different readers, it is different for the number of reading books at a certain time, that is, the score cannot be calculated only according to 60 days, but also need to introduce the concept of average borrowing, at this time, \(t_{book}\) in formula (6) needs to be changed to.

\[
t_a = \frac{1}{\left|D(a)\right|} \sum_{i \in D(a)} t_{ai}
\]  

(7)

In Formula (7), \(t_{ai}\) denotes the borrowing time of user A to book i, \(D(a)\) denotes the total number of books borrowed by user A, that is, how many books borrowed by user A, and \(t_a\) denotes the average borrowing time of user A. By substituting \(t_a\) and \(t_{ai}\) into the formula (6), the book lending scoring formula in this paper is obtained.

\[
S_{book} = \begin{cases} 
10, & t_{ai} \geq 2t_a \\
\frac{t_{ai}}{t_a} \times 10, & t_{ai} < 2t_a 
\end{cases}
\]  

(8)
3.2. Improved Item Scoring Method
The similarity method between users is applied to the scoring calculation of items. Taking books as an example, the number of books ranges from hundreds of thousands to tens of millions, in order to traverse quickly, the improved cosine similarity method mentioned above is adopted. At the same time, because the number of projects is large and most projects have no similarity, N items with the highest similarity can be selected from the project set as the initial project set. From the perspective of books, that is to say, finding N books that are more relevant to users' interests, they can be expressed as \( U_b = \{q_1, q_2, q_3, ..., q_n\} \), among them, \( q_1 \) represents the book with the highest similarity to users, and \( q_n \) represents the book with the lowest similarity to users.

When the user-book scoring matrix is sparse, the data items of common scoring may be too few, and the result obtained by Pearson correlation coefficient is unreasonable. Assuming that there are users A and B, their combined scoring set can be used as the common scoring data of two users, and then the common scoring data items among users are added, among them, the set of scoring items of user A is represented as \( E_A \), and the set of scoring items of user B is represented as \( E_B \), then the common scoring sets of user A and user B are expressed as.

\[
U_{AB} = E_A \cup E_B
\]  

Taking User A as an example, the item set scored by the user is the key to the prediction scoring algorithm, because the result of the prediction scoring is directly related to whether the neighbor objects with similar characteristics of the target user can be found correctly, user A's non-scored items can be expressed as follows.

\[
C_A = U_{AB} - E_A
\]

Extract any item \( d \) from item set \( Y_i \), and get the neighbor set \( Cd \) of the item, according to this set, user A's predictive score \( T_{ad} \) for item \( d \) can be solved, \( sim(d, u) \) represents the similarity between item \( d \) and item \( u \), \( R_{a,u} \) represent user A's actual score for item \( u \), the formula is as follows.

\[
T_{ad} = \frac{\sum_{u \in Cd} sim(d, u) \cdot R_{a,u}}{\sum_{u \in Cd} sim(d, u)}
\]  

In formula (11), if item \( d \) has no similar object, the result of \( sim(d, u) \) corresponding to all items \( d \) and \( u \) is 0, at this time, the predictive score \( T_{ad} \) of user A for item \( d \) can be expressed as the average score value of user. The core idea of the whole algorithm is to take the union of items as the default values of different items, at the same time, in order to get a larger common scoring data set, it needs to merge multiple items, the more items are merged, the greater the consumption of calculating different default values, that is to say, the complexity of the algorithm is very large, and it can reach \( O(n^2) \).

4. Personalized Book Recommendation Based on Improved Collaborative Filtering Algorithm
The traditional collaborative filtering algorithm and improved filtering algorithm are used in the book recommendation system. Firstly, the implementation process of the system is introduced. Then, the experimental data sources of this study are explained. Finally, the effect of book recommendation is compared.
4.1. The Execution Process of Book Recommendation System

![Flow chart of recommendation system](image)

Figure 1. Flow chart of recommendation system

After users log in to the system, they need to check the user's level, after reaching a certain level, the system can extract the historical data that user access and find the type of data they choose. According to the type of books, the system adopts collaborative filtering algorithm to discover the neighbors with similar interests of the users. It calculates the scoring results of neighbor books and recommends popular books to users.

In the process of recommending books, we need to consider the evaluation of books and adopt collaborative filtering algorithm, which can screen out books that users are interested in, but the number should not be too large, and users are more interested. At the same time, user rank can be set according to the reading level of books, and the implementation process is shown in Figure 1.

4.2. Data Source of Book Recommendation System

By introducing the traditional collaborative filtering algorithm and improving the collaborative filtering algorithm, we compare the performance of the system before and after using, and get book data from Book Crossing website, which is a sharing website, and users can tag their favorite books when they enter the website, so as to facilitate their own search.

Users of the website reaches 175W and the total number of books reaches more than 18 million, in the experiment, we select user access data retained from the web site in July 2017, the number of users is 256532, the number of books is 207487, and the number of scoring records reaches 11754720, among them, most of the data has been cleaned up, there are a certain number of invalid records, and some data has the question of irregular formats. User ratings are divided into two categories, and some users give an evaluation to the selected books without scoring, this kind of evaluation score is set to 2 points. Some users directly score the selected books, the maximum score is set to 10 points, and the popularity of the books can be set according to the score.

4.3. Implementation of Book Recommendation System

Adopting the traditional historical scoring method, that is, traditional collaborative filtering method is adopted, the recommended results are filtered from the predictive scoring records, arrange them from big to small, and the data set is Book-Crossing, the data set is big, the number of users is big, and the execution efficiency of the algorithm is lower, as shown in Figure 2.
In Figure 2, we can see that many unrelated results are recommended, because the user is a computer-oriented user, the recommended records include "Interview with the Vampire" and "The Da Vinci Code", these are fictions, not technical books.

An improved collaborative filtering recommendation algorithm is adopted, the algorithm introduces a project-based common scoring method to recommend books, the results are shown in Figure 3.
In Figure 3, we can see that there are many effective recommendation results on the page, the user's interest is in the Java Web domain, and the recommendation results exclude many invalid recommendations, the effective recommendation books include "Java Development 1800 Examples", "Java and Struts 2.0", "Java and Spring" and other books, these books are all books on Java Web development, that is, the recommendation results is in line with the user's interest.

5. Conclusion
This paper studies recommendation algorithm, we find that the invalid recommendation results are reduced and the effective recommendation results are increased, that is, the algorithm is effective and can screen out neighbor results similar to users' interests.

6. Acknowledgement
The authors would like to thank for financial support by youth fund project of the humanities and social sciences of Education Ministry (15YJC870004), science and technology innovation team of XiangNan University, Hunan province undergraduate research-based learning and innovative experimental project (719), social science planning project of Chenzhou (Czsskl2017067) and Institute of big data science of XiangNan University.

7. Reference
[1] Kechao, L. I ., & Zhengyou, L. . (2012). Personalized book recommendation algorithm based on multi-feature. Computer Engineering, 38(11), 34-37.
[2] Zhang, F. . (2016). A personalized time-sequence-based book recommendation algorithm for digital libraries. IEEE Access, 4, 2714-2720.
[3] Xiangyun, Z. , Zhigang, C. , Rui, H. , Bo, L. I. , Software, S. O. , & University, C. S. . (2015). Personalized book recommendation algorithm based on topic model. Journal of Computer Applications.
[4] Pathak, D., Matharia, S., & Murthy, C. N. S. . (2013). NOVA: Hybrid book recommendation engine. Advance Computing Conference. IEEE.
[5] Breese, J. S. . (1998). Empirical Analysis of Predictive Algorithms for Collaborative Filtering. Proc. of the 14th Conference of Uncertainty in Artificial Intelligence, Dallas, Texas, San Francisco, CA, Morgan Kaufmann, 1998.
[6] Su, Xiaoyuan, Khoshgoftaar, & Taghi, M. . (2012). A survey of collaborative filtering techniques. Advances in Artificial Intelligence, 2009(12), 4.
[7] ChangyongLiang, & YajunLeng. (2014). Collaborative filtering based on information-theoretic co-clustering. International Journal of Systems Science, 45(3), 9.
[8] Wu, J., Chen, L., Feng, Y., Zheng, Z. , Zhou, M. C. , & Wu, Z. . (2013). Predicting quality of service for selection by neighborhood-based collaborative filtering. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 43(2), 428-439.
[9] Choi, K., Yoo, D., Kim, G., & Suh, Y. . (2012). A hybrid online-product recommendation system: combining implicit rating-based collaborative filtering and sequential pattern analysis. Electronic Commerce Research & Applications, 11(4), 309-317.