



Algorithm of Collaborative Filtering Recommendation and Its Application in Electronic Shopping Mall

Jianqiao Shen¹(✉) and Tianwei Zhu²

¹ Department of Computer and Information Science, DongGuan City College,
DongGuan 523419, China
stories53.student@sina.com

² DongGuan City College, DongGuan 523419, China

Abstract. According to the different objects concerned in collaborative filtering recommendation algorithm, it is divided into user-based and item-based collaborative filtering recommendation algorithm. This article analyzes and designs two algorithms according to the principle and recommendation functions. And can analyses the users' potential consumer markets by applying these collaborative filtering recommendation algorithms in vertical e-commerce malls. The results show that the effect of collaborative filtering algorithm is obvious, and it has good performance compared with traditional algorithms.

Keywords: Collaboration · Recommendation algorithm · Filter · E-commerce mall

1 Introduction

It is difficult for consumers to find products which are truly suitable for them in an assortment of commodity world. It is necessary to use a recommendation system of the electronic mall website to automatically judge the products which have actual demands for consumers according to the consumers' past consumption records and browsing records. The essence of the recommendation system is the recommendation algorithm, which can infer the items that the user might like. There are many types of recommended algorithms, and collaborative filtering is one of the most widely used personalized recommendation technologies. Collaborative filtering is mainly divided into user-based collaborative filtering and item-based collaborative filtering.

This paper designs two kinds of algorithms based on the user's collaborative filtering recommendation algorithm and the item-based collaborative filtering recommendation algorithm by introducing the application of a vertical electronic shopping mall website—the watches network mall. It designs collaborative filtering recommendation algorithms and applies these algorithms to the recommendation of products.

2 Collaborative Filtering Recommendation Algorithm

2.1 User-Based Collaborative Filtering Recommended Algorithm

User-based collaborative filtering is the most widely used method in personalized recommendation. It predicts the interest preference of target users based on the interests of neighbor users. The algorithm first uses statistical techniques to find neighbors that have the same preferences as the target user, and then generates recommendations to the target users based on the preferences of the neighbors of target users.

Its basic principle is to use the similarity of user access behavior to recommend resources that users may be interested in. The system calculates the closest users of number N with its access behavior (such as purchased product collection, visited web page sets, etc.) by using a specific similarity function as the nearest neighbor collection. The nearest users of number N combine as the nearest neighbor set of the user, and the neighboring users of the statistics access the resource to generate a candidate recommendation set. Finally calculates the recommendation degree of each resource in the candidate recommendation set to the user, and take the top-K as the recommended set of the user (Fig. 1).

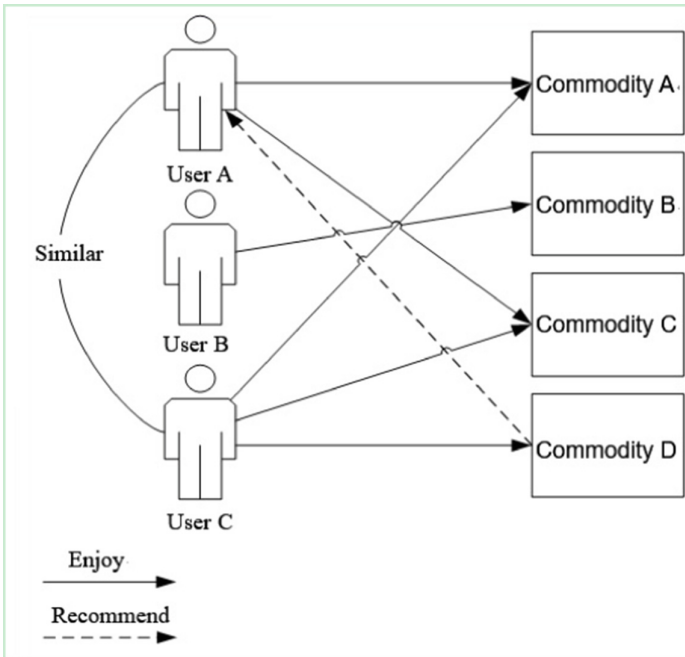


Fig. 1. Example of user-based collaborative filtering recommendation algorithm.

As shown in Formula (1), the favorite items of user A are A and C, the favorite item of user B is B, and the favorite items of user C are A, C and D. Using the cosine similarity algorithm to calculate the similarity between users, we get the result that the user A and

user C have the highest similarity. Then find the most similar n users from the user set and extract all the products that the user likes, and remove the user’s own favorite item. For each candidate item, the degree to which the user may be interested in is calculated by a specific formula as shown in Formula (1).

$$p(u, i) = \sum_{v \in S(u, K) \cap N(i)} w_{uv} r_{vi} \tag{1}$$

In this formula, u is the target user, S is the item set, and r_{vi} is the degree of preference of user v to user i . In a similar system, the user score is substituted, and this user score is formed by user behavior modeling. Finally, according to the neighboring users, the first few items are sorted according to the preference level.

2.2 Item-Based Collaborative Filtering Recommendation Algorithm

The basic idea of the item-based collaborative filtering recommendation algorithm is to calculate the similarity between items based on historical behavior data of all users in advance, and then recommend the items similar to the items that the user likes to the user (Fig. 2).

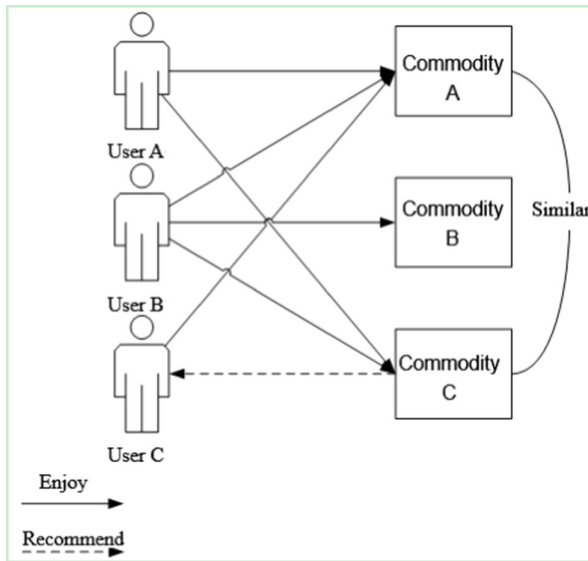


Fig. 2. Example of item-based collaborative filtering recommendation algorithm recommendation.

As shown in Formula (2), user A has acted on items A and C. User B has acted on items A, B and C, and user C has acted on item A. So we can get that the number of users who both like items A and C is 2. Item A is similar to item C, and then calculate the user's interest of the item by the formula in Formula (2).

$$p(u, j) = \sum_{i \in N(u) \cap S(i, K)} w_{ji} r_{ui} \quad (2)$$

In this formula, u is the target user, N is the collection of items that the user likes, S represents the set of K items most similar to item i , w_{ji} represents the similarity of item j and item i , and r_{ui} represents the interest of user u for item i .

3 Implementation of the Collaborative Filtering Recommendation Algorithm

3.1 Introduction and Design of Recommended Products Function

In an assortment of commodity worlds, it is often difficult for consumers to find products that are truly suitable for them. Therefore, the recommendation system or recommended function of the e-shop website has emerged. This article takes an electronic shopping mall website with watches as its main sales item as its background. One of its featured functions is to recommend the items that users may like to consumers. By adopting the collaborative filtering recommendation algorithm, it quickly guesses the customer's consumption intention and improves the convenience of customer's browsing, while increasing the sales of goods. It is mainly divided into user-based collaborative filtering and item-based collaborative filtering.

First, a brief introduction of the function of the collaborative filtering recommendation algorithm using in the electronic mall is given. It is mainly a "commodity recommendation" module. The product recommendation module mainly uses a user-based collaborative filtering recommendation algorithm and an item-based collaborative filtering recommendation algorithm to implement a function of personalizing recommended products to each user. The detailed description is shown in Table 1.

Table 1. Product recommendation module description.

Module number	Module name	Product recommendation
Module function description		<p>The product recommendation module mainly implements the function of personalizing recommended products to each user, and uses the user-based collaborative filtering recommendation algorithm to implement the “personalized recommended products” of the home page; using the item-based collaborative filtering recommendation algorithm to implement the “Guess what you like” in product details page:</p> <p>(1) The user-based collaborative filtering recommendation algorithm recommends the product: the system records the user’s behavior data, and derives the user’s preference for the product according to the weight score of the behavior, the recommendation algorithm finds a similar user set according to the user’s preference for the product. Then, according to other products that similar users like, organize it into a directory and recommend it to users</p> <p>(2) Item-based collaborative filtering recommendation algorithm recommends item: the algorithm calculates the similarity between items by the user’s historical rating behavior of the item, and then recommends the item similar to the item that the user likes to the user</p>
Remarks		<p>(1) Record of user behavior: set up a user behavior table and set the corresponding behavior field (browse, forward, collect, purchase), the behavior field takes 0 and 1 for recording, 0 means the user has no corresponding behavior for the product, 1 represents the user in the corresponding behavior of the product</p> <p>(2) Calculate the user’s preference of the product: use the trigger of the database, and write a method for calculating the user’s preference for the product according to the behavior weight. When the user triggers the behavior of the product, the trigger automatically updates the user’s degree of love of the product according to the prepared method</p> <p>(3) Implementation of collaborative filtering recommendation algorithm: using the user-based collaborative filtering recommendation algorithm (UserCF) and the item-based collaborative filtering recommendation algorithm (ItemCF) in the Apache Mahout open source project to complete the product recommendation</p>

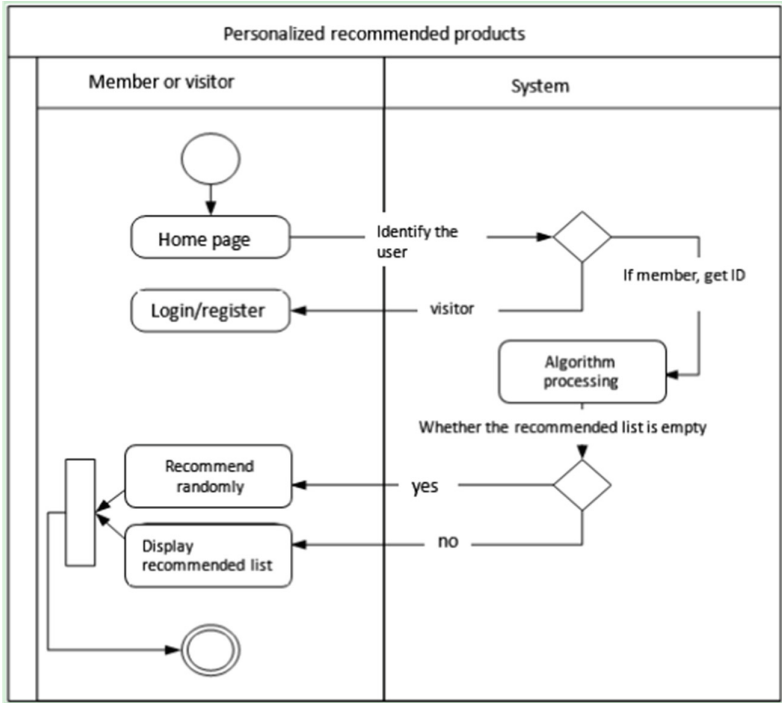


Fig. 3. Personalized recommended product activity chart.

Collaborative filtering recommendation algorithm recommended product function activity design diagram shown in Fig. 3.

The user enters the home page of “Amoy Online” to determine whether the user identity is a registered member. The system recommends the product for each member user in the “Personalized Recommended Products” section of the homepage and “Guess You Like” in the product details page. The system identifies the user identity, and the member id number is processed by the algorithm to obtain the recommended product list. Due to the recommendation mechanism, the recommended product list may be empty. If the recommended product list is empty, the member user is randomly recommended. The corresponding activity diagram and timing diagram of the product are shown in Fig. 4.

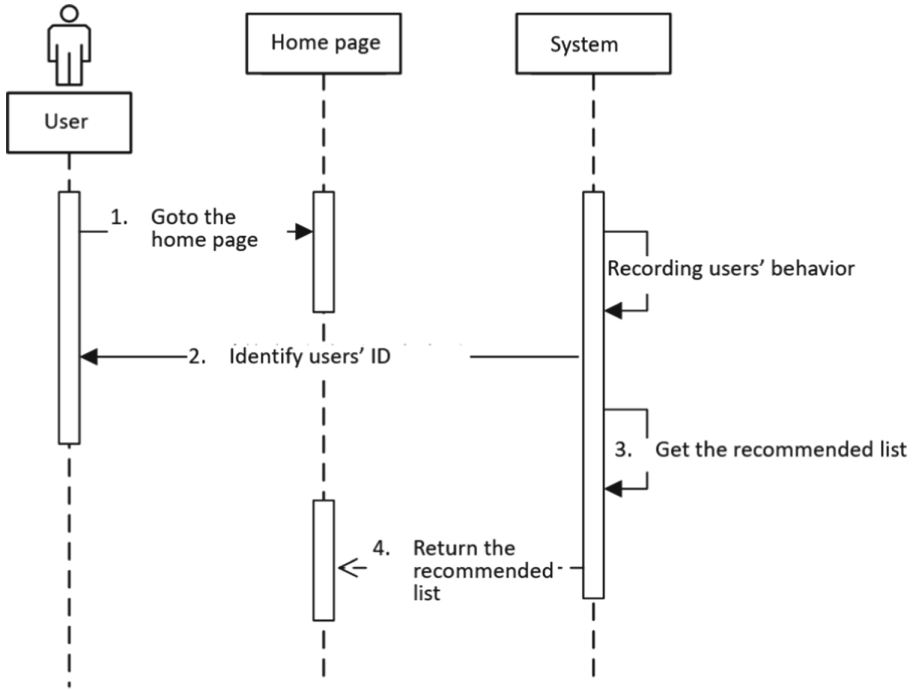


Fig. 4. Personalized recommended product timing chart.

3.2 User-Based Collaborative Filtering Recommendation Algorithm Recommended Products

In many cases, users are not really clear about their needs, or their needs are difficult to express with simple keywords. Or they need to be more in line with their personal tastes and preferences. Therefore, using the user-based collaborative filtering recommendation algorithm to learn from other preferences of other similarly high-profile member users, it is possible to personalize the recommended products for each member user, and complete the relevant recommendation of the homepage in the mall. As shown in Fig. 5.

- (1) The system records the browsing, forwarding, collecting, purchasing and scoring behaviors through the recordBrowse(), recordRepeat(), recordCollect(), recordBuy(), and recordMarking() methods of the BehaviorController control class, passing in the user id and the product id. Each behavior is tagged, with browse, forward, favorite, and purchase using 0 or 1 to flag whether this behavior has occurred, with a score of 0 to 5 being scored.
- (2) When the user enters the mall page, the system performs the personalized product recommendation by calling the getRecommendGoods() method of the GoodsController control class. First, it is judged whether the user is a member user through the getAttribute() method of the session. If it is the member user, the user id will get into the getRecommendedGoods() method in the UserBased class to get the user's personalized recommended product list. Finally the product data is encapsulated into

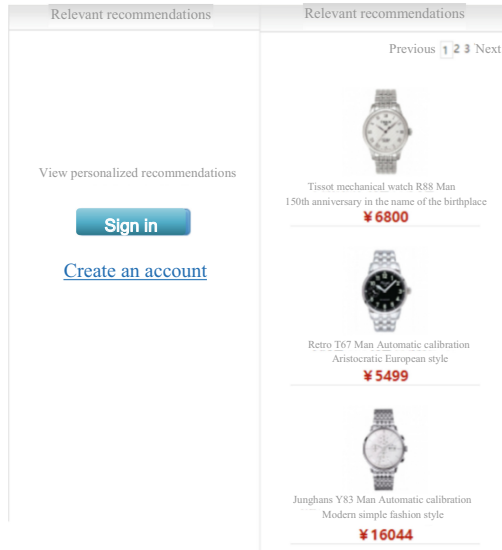


Fig. 5. Personalized recommendation bar before login and personalized recommendation bar after login.

json data and returned to the foreground to complete the product recommendation. If the user is a newly registered user or the user behavior data is too small to generate a recommended product list, the merchant product is randomly recommended to the user.

- (3) The personalized recommendation is mainly based on the user's behavior to derive a preference for a certain product, and the user-based collaborative filtering recommendation algorithm is substituted to obtain a personalized product recommendation list for each user. Users' behaviors include browsing, forwarding, collecting, purchasing and scoring. The system sets its behavior weights to 5%, 15%, 20%, 30%, and 30% respectively. Use the triggers in the database to calculate the degree of preference and update the preference field in the user behavior table based on the user's behavior for each item as a trigger point.
- (4) Create a DataModel containing the user id, the product id and the user's preference for the product. Pass the model into the PearsonCorrelationSimilarity() method of Apache Mahout. To calculate the Pearson similarity to get the similarity between users, and pass the returned result to NearestNUserNeighborhood. The () method creates an instance of a neighbor user, and finally the recommended() method in the GenericUserBasedRecommender class gets the recommendation result by calling getTopItems() of the TopItems class.

3.3 Item-Based Collaborative Filtering Recommendation Algorithm

The mall system uses the item-based collaborative filtering recommendation algorithm to calculate the similarity between products by rating the history of the product, and then

generating a recommendation list for the user according to the similarity of the item and the historical behavior of the user, which is completed in the mall. The product listing page guesses the recommendation of the column you like, as shown in Fig. 6.



Fig. 6. Product details page guess you like the column to display recommended product results

- (1) When user confirms the receipt, the system will let the user rate the product, and the content of the rating will be recorded in the comment table. The information in the table includes the user id, the product id, and the user's comments to the product.
- (2) When users click on the products' detail page, the system will grab the user's session information, get its user id and pass it to the `getRecommendedGoods()` method of `ItemBased` which is an implementation class to obtain the recommendation list.
- (3) Create a `DataModel` in the `getRecommendedGoods()` method, encapsulate the user id, product id, and score in the comment table with the `DataModel`, and then pass the model to the `PearsonCorrelationSimilarity()` method of `Apache Mahout` to calculate the Pearson similarity. The similarity between the items, finally pass the similarity information of the model and the item into the `GenericItemBasedRecommender` class. It calls the `recommended()` method in the class and call the `getTopItems()` of the `TopItems` class to get the recommendation results.

4 Summary of Collaborative Filtering

Here gives the comparison of advantages and disadvantages between UserCF (User-based collaborative filtering recommendation algorithm) and ItemCF (Item-based collaborative filtering recommendation algorithm), as shown in Table 2.

Table 2. Comparison of advantages and disadvantages between UserCF and ItemCF

	UserCF	ItemCF
Performance	It is suitable for occasions with few users. If there are many users, it is very expensive to calculate the user similarity matrix	It is suitable for situations where the number of items is significantly less than the number of users. If there are many items (web pages), it is very expensive to calculate the item similarity matrix
Real time	Real time users have new behaviors, which may not necessarily lead to immediate changes in recommendation results	If users have new behaviors, they will certainly lead to real-time changes in recommendation results
Cold boot	After a new user acts on a few items, he cannot be personalized recommended immediately, because the user similarity table is calculated offline at regular intervals	As long as a new user acts on an item, he can recommend other items related to the item
Reasons for recommendation	It is difficult to provide convincing recommendation explanations	It is difficult to provide users with convincing recommendation explanations. Users can be more convinced by using users' historical behavior to make recommendation explanations to users

As a classic recommendation algorithm, collaborative filtering algorithm is widely used in industry. It has many advantages, and the model has strong versatility. It does not require much professional knowledge in the data field. The project is simple to implement and the effect is obvious. These are the reasons why it is popular. Of course, collaborative filtering has some inevitable problems, such as the headache of "cold start". When we don't have any data from new users, we can't recommend items for new users. At the same time, the differences in the situation are not considered, such as the users scene and the users current mood. There is no way to get some niche's unique preferences. This piece is based on content-based recommendations.

Acknowledgement. This paper is supported by the Dongguan social science and technology development (key) project, ID: 2020507151144. This work is supported by Key Field Special Project of Guangdong Provincial Department of Education with No. 2021ZDZX1029.

References

1. Yan, Y.: Research on the design of shopping website interface based on user experience. *China Packag. Ind.* **11**(16), 19–20 (2014)
2. Sun, W.: *Java Object-Oriented Programming*, 2nd edn. Publishing House of Electronics Industry, Beijing (2017)
3. Parmar, M.D., et al.: FREDPC: a feasible residual error-based density peak clustering algorithm with the fragment merging strategy. *IEEE Access* **7**, 89789–89804 (2019)
4. Anonymous: Online clothing mall shopping system based on SSM framework. Guangdong University of Technology (2016)
5. Zaheer, M.Z., Mahmood, A., Astrid, M., Lee, S.-I.: CLAWS: clustering assisted weakly supervised learning with normalcy suppression for anomalous event detection. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) *ECCV 2020*. LNCS, vol. 12367, pp. 358–376. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58542-6_22
6. Xiao, R.: *SSM Enterprise-Level Framework Combat*. Water Resources and Hydropower Press, Beijing (2017)
7. Wan, B., Fang, Y., Xia, X., Mei, J.: Weakly supervised video anomaly detection via center-guided discriminative learning. In: *Proc. IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6, July 2020
8. Zou, S.: *UML Object-Oriented Requirements Analysis and Modeling Course*. Science Press, Beijing (2017)
9. Liu, Z.: *MySQL 5.7 from Entry to Proficiency*. Tsinghua University Press, Beijing (2016)
10. Zhang, Z., Chen, D., Wang, Z., Li, H., Bai, L., Hancock, E.R.: Depth-based sub-graph convolutional auto-encoder for network representation learning. *Pattern Recogn.* **90**, 363–376 (2019)
11. Zhu, Y., Newsam, S.: Motion-aware feature for improved video anomaly detection. In: *Proceedings of British Machine Vision Conference* (2019)
12. Zhang, X.: *CSS World. People's Posts and Telecommunications Press*, Beijing (2017)
13. Hou, L., Nguyen, V., Kanevsky, A.B., et al.: Sparse autoencoder for unsupervised nucleus detection and representation in histopathology images. *Pattern Recogn.* **86**, 188–200 (2019)
14. Parmar, M., et al.: REDPC: a residual error-based density peak clustering algorithm. *Neurocomputing* **348**, 82–96 (2019)
15. Tomski, A.: AnnoGene: RESTful web service for annotating genomic features. *J. Appl. Comput. Sci. Methods* **6**(2), 101–110 (2014)