I J C T A, 8(5), 2015, pp. 2043-2049
© International Science Press

Design of A Recommendation System Using Hybrid Collaborative Filtering

Samundeeswary K* and Vallidevi Krishnamurthy**

Abstract: Recommender system's (RS) goal is to make personalized recommendations of items/products by using filtering algorithms like collaborative filtering (CF), which helps users to select appropriate items from large dataset. CF predicts the utility of an item for a particular user based on user's previous interest or by taking opinions from other users. The purpose of RS is to provide personnel assistance to the customers to find out the best items out of most used ones or the best item which has maximum popularity. The vast growth of products and users over internet in recent years poses some key challenges in recommender system. The greatest challenge is to avoid data sparsity and cold-start problem and to produce high quality recommendation. We build a recommender system using combined result of user based/item based collaborative filtering algorithms, inorder to increase the accuracy and quality for making predictions and recommendations of items to the users. Movielens dataset is used in this recommendation systems.

Index Terms: recommender systems, collaborative filtering, scalability, sparsity, cold start problem**.**

1. INTRODUCTION

The growth of the websites has rapidly increased the products over the internet. This makes the consumers, unable to find the relevant item they want to purchase through the internet. Consumers need some assistance in exploring new product which leads to the need of recommender system. There are various reasons for business people making use of recommender systems [4].To increase the number of items consumed by the user: In an e-shopping business which may increase the number of items or products purchased by the user.

To enable the consumers, select the items that may be hard to find without a precise recommendation: This functionality enables non-popular items and items that the users do not usually consume to come in the notice of the users to purchase.

To increase the consumer's satisfaction: A well designed recommender system finds the consumers interest based upon their search.

To better understand the consumer's interest: From the consumer's preferences, either collected explicitly or implicitly predicted, service providers and business managers may build new marketing policies that suits user's tastes on an individual level.

Collaborative filtering is the technique used to find the recommendations for the consumers. This filtering technique is based on two different types of algorithms [2].

Memory-based algorithms make use of the entire user-item database to generate a prediction. These systems use the statistical techniques to find a set of users, known as *neighbors.* These *neighbors* have a history of agreeing with the target user. Thus they rate different items similarly or they tend to buy similar

* PG Student, Department of Computer Science and Engineering, SSN College of Engineering, Chennai, India, *Email:samundeeswary93@gmail.com*

** Assistant Professor, Department of Computer Science and Engineering, SSN College of Engineering, Chennai, India, To better understand the consumer's interest: From the consumer's preferences, either collected explicitly or implicitly predicted, service providers and business managers may build new marketing policies that suits user's tastes on an individual level, *E-mail:vallidevik@ssn.edu.in*

sets of items [7]. Once a neighborhood of users is identified, these systems use different algorithms to combine the preferences of neighbors and thus produce a prediction or *top-N* recommendation for the active user [2]. The techniques *nearest-neighbor* or user-based collaborative filtering are widely used and also more popular. In the context of actual systems that generate real-time recommendations on the basis of very large datasets, memory-based recommendation systems are not always as fast and scalable as we expect them to be. Hence, to achieve these goals, model-based recommendation systems are used.

Model-based recommendation systems [6] involve building a model based on the dataset of ratings. In this method, some information from the dataset are extracted, and used as a "model" to make recommendations in-order to avoid using the complete dataset every time. This approach provides the benefits of both speed and scalability. Collaborative filtering (CF) can be achieved using User Based Recommendation or Item Based Recommendation.

2. RELATED WORKS

Recommender system produce list of recommendation information, either using collaborative filtering or content-based filtering. Recommender systems built using content-based algorithms, which makes use of user's profile information such as has information about a user and his taste are recommender system built at the beginning [1]. Recommender engine compares the items that were already rated positively by the user with the items, that were not rated and looks similar to the positively rated items that will be recommend to the user.

Collaborative filtering [2] algorithms based recommender systems, idea is to identify the user in a community that shares similar interest. If two users have same or almost similar rated items in common, then they are identified as the user sharing similar tastes. Such users are built as a group called as neighbourhood. A user gets recommendation to those items that he/she hasn't rated before, but that were already positively rated by user in his/her neighbourhood. Empirical Analysis of Predictive Algorithm for Collaborative Filtering by J. S. Breese [2], can easily take new users ratings into account were it cannot cope well with large number of users and items, since their online performance is often slow.

User based collaborative filtering [3], where the majority of users sharing common interest are joined into a group. If the items are positively rated by the community those items will be recommended to the remaining members of the group. The benefit is that as the data are straightforward, the system doesn't need any complex functions to analyze the data.

There are some disadvantages of user-based collaborative filtering Firstly, when there are less number of users in the system at the beginning time, then the opinion could be biased by the existing users. At that time systems could make wrong expectation for new users. Secondly, the system only depends on preference ratings, which may cause first-rate and cold-start problems.

WSrec: a collaborative filtering based web service recommender System [4], which includes a user-contribution mechanism for web service Qos value prediction. The user-based and item-based CF algorithm are used to recommend web services, since the two approaches recognized the different characteristic between web service QoS and user ratings, the prediction accuracy of these methods was unsatisfactory.

L.H. Ungar and D.P. Foster [5], clustering model treats collaborative filtering as a classification problem and works by clustering similar users in same class and estimating the probability that a particular user is in a particular class, and from there computes the conditional probability of ratings. Benefit of using clusters is to reduce the number of items and users, while making suggestion and by solving the large computational problem however, this will reduce the quality of recommendations. In other words, if the method will compare the user to a small sample, the similarity will not be accurate. Also, partitioning items to item-space will limit the recommendations to specific types of products. Additionally, if the user already bought these items, then they will never be recommended to him/her.

Item based filtering [6], items are rated and used as parameters instead of users. Item-based algorithm mainly depends on the relationships between items. The system discriminates between interesting or uninteresting products for the user by examining the user's purchase history. After then, recommendations for users are computed by finding items that are similar to other items the user has liked. This technique is well known by Amazon.com. In the Amazon.com, the systems recommends items to users by showing the message, "users who bought something also bought some other things". Item-to-item CF in Amazon [6], follow the iterative algorithm that provides a better approach by calculating the similarity between a single product and all related products. It is possible to compute the similarity between two items in various ways, but a common method is to use the cosine measure, in which each vector corresponds to an item rather than a customer, and the vector's M dimensions correspond to customers who have purchased that item. Therefore, it has been said that item-based CF is more time efficient than user-based (memory-based) CF.

Manos Papagelis and Dimitris Plexousakis [7], utilize explicit ratings in an 'implicit' sense, so as to enrich a user's model, without actually prompting users to express their preference to categories. Thus, item-based prediction algorithm with explicit rating provides better accuracy.

Yueping Wu and Jianguo Zheng [8] work is based on improved similarity measure method which automatically generates weightage factor to combine dynamically item attribute similarity and score similarity. Using item similarity, recommendation system determines nearest neighbor of item. This predicts the item's rating for recommending items to users.

SongJie Gong [9] builds recommendation system that combines user clustering and item clustering technology where users are grouped based on objects ranking. It suffers from its poor quality, when the number of records in the user database increases.

Simon Renaud-Deputter, Tengke Xiong, Shengrui Wang [10] makes use of only implicit feedback on user purchase made on the past to discover the relationships within the users. Based on the clustering results, products of high interest were recommended to the users using high-dimensional parameter-free clustering, where implicit feedback does not always provide sure information about the user's preference.

3. PROPOSED SYSTEM

This paper focuses on building a recommendation system using Collaborative Filtering (CF) and clustering techniques. The proposed work has 3 main modules. (i) Apply User-based CF. (ii) Applying Item-based CF. (iii) Combine the results of user-based CF and item-based CF.

4. SYSTEM ARCHITECTURE

Figure 1 represents the overall system architecture. The detailed procedure for the proposed system is explained in the module description section given below. Dataset used here is *movielens* of 100,000 rating (1-5) for 1682 movies by 943 users, where, each user has rated atleast 20 movies. Data preprocessing is a data mining technology that involves transforming raw data into an understanding format. Preprocessed the data are converted into user-item matrix.

5. MODULE DESCRIPTION

5.1. User Based Collaborative Filtering

User based CF algorithm works on the assumption that each user belongs to a group of similar behaving users. The basics for the recommendation are composed by items that are liked by users. Items are recommended based on user's interest (preference on items). The algorithm considered here is that users with similar interest will be categorized as same items. Figure 2 (a) explains the steps involved in User based CF.

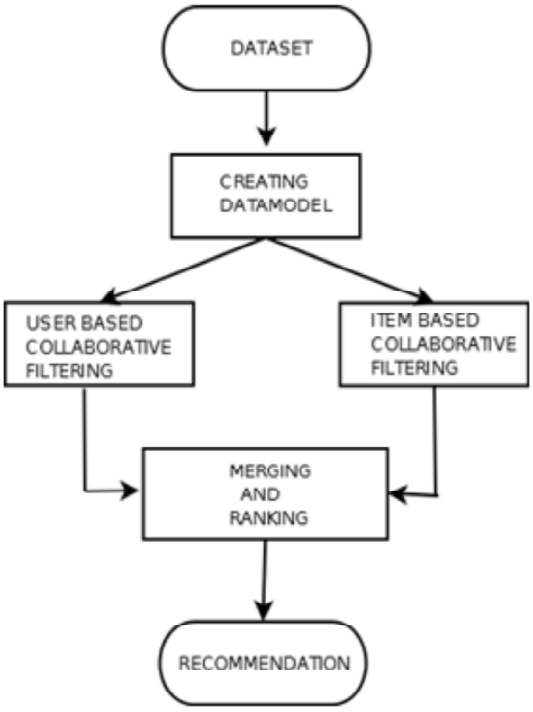


Figure 1: System Workflow

(i) Similarity Computation for Nearest Neighbour

The similarity between the user '*v*' and active user '*au*' has to be calculated in-order to find the nearest neighbor where '*r*' is the rating and '*it*' is the item. Cosine similarity formula is used to compute the similarity between two users.

$$sim(au, v) = \cos(\overrightarrow{r_{au}}, \overrightarrow{r_v}) \tag{1}$$

$$= \frac{\overrightarrow{r_{au}}, \overrightarrow{r_v}}{\left\|\overrightarrow{r_{au}}\right\|_2 * \left\|\overrightarrow{r_v}\right\|_2} \tag{2}$$

$$= \frac{\sum r_{au,it}, r_{v,it}}{\sqrt{\sum r_{au,it}{}^2} \sqrt{\sum r_{v,it}{}^2}} \tag{3}$$

(ii) Find the Nearest Neighbours

After calculating the similarity between each user with the active user, the computed similarity values are used in selecting *K* users, having highest similarity with the active user. The KNN (*K* Nearest Neighbour) algorithm predicts the item that an active user may like to purchase.

(iii) Calculated Weighted Average

The items are predicted based on the weighted average of deviation from the neighbor's mean as in

$$Pred_{au,it} = = \vec{r}_a + \frac{\sum_{u=1}^{k} (r_{au,it} - \vec{r}_{au}) * sim(au, v)}{\sum_{u=1}^{k} \sin(au, v)} \tag{4}$$

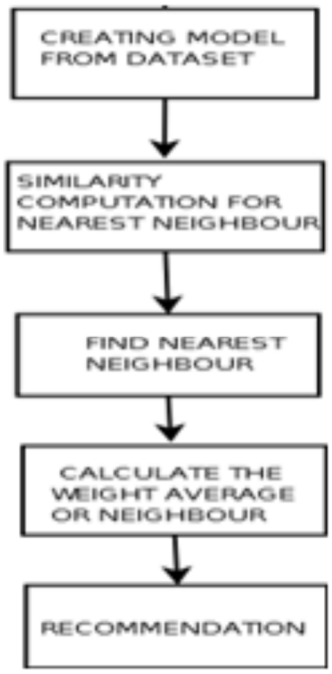
Where $Pred_{au,\,jt}$ is the prediction for the active user 'au' from item 'it'.

(iv) Top N item Recommendation

The predicted value for each item purchased by the nearest neighbor of active user has been calculated. Based on the predicted rating value, the items are grouped and the top *N* items with highest predicted rating are recommended to the user.

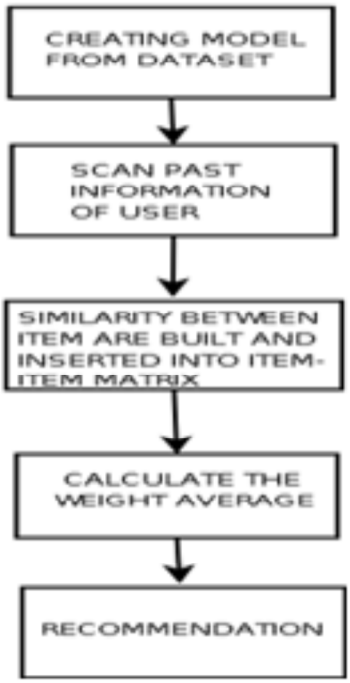
5.2. Item Based Collaborative Filtering

Item based recommender algorithms look at the similarity between items to determine the prediction. The idea is that, the user is most likely to purchase items that are similar to the one the user already has purchased in the past. This is done by analyzing the purchasing information of the user, either explicitly (by rating) or implicitly (user browsing information). Figure 2 (b) explains the steps involved in Item based CF.



(a)

Figure 2 (a): Design of User Based Collaborative Filtering



(b)

Figure 2 (b): Design of Item Based Collaborative Filtering

(i) Scan Past Information of User and Similarity Calculation

The past purchase information of the user, i.e. the rating they gave to items are collected. With the help of this information the similarities between the items are built and inserted into 'item to item' matrix. Items which are most similar to the items rated by the user in the past are selected and the cosine similarity is used to compute similarity between the items.

$$sim(it, jt) = \cos(\vec{it}, \vec{jt}) \tag{5}$$

$$= \frac{\vec{it}.\ \vec{jt}}{\left\|\vec{it}\right\|_2 * \left\|\vec{jt}\right\|_2} \tag{6}$$

(ii) Calculated Average Weight

The weighted average for items are predicted using the past item ratings, by using the formula given below. The prediction is done between the user '*v*' and item 'it' ' using the similarity calculated between items 'it' and 'jt'.

$$Pred_{v,it} = \frac{\sum all\ similar\ items,\ N\left(r_{au,jt} * sim\left(it, jt\right)\right)}{\sum all\ similar\ items,\ N\ |\ sim(it, jt)\ |} \tag{7}$$

(iii) Top N Item Recommendation

The predicted value for each item has been calculated. Based on the predicted rating, the items are grouped and the top *N* items with highest predicted rating are recommended to the user.

5.3. Combined Result of Item Based and User Based CF

In case of user-based CF, if the nearest neighbors (similar users) are not in enough number, i.e. interest of target users are not similar to many users, then recommending an item for that particular user may not be accurate. Item-based CF is based on the past information of the user, so it works well in such cases. As the main objective of this work is to satisfy the users, the user based CF and the item based CF which has high speed compared to user based are combined. The results of user and item based CF are merged and based on the predicted rating value on each item, clustering and ranking of items are performed. Thus the top ranked items are recommended to the users.

6. EXPERIMENT AND EVALUATION

Experiment and evaluation is one of the most important parts in proposing new algorithm since it is the way to objectively assess the effect of the improved algorithms.

6.1. Datasets

To evaluate the new algorithm in this paper, Movielens 100k dataset is used here. Movielens was published by Grouplens project team of Minnesota University in America. 1682 users rank 943 movies that they've ever seen and scoring range is 1-5 points. It also contains the timestamp for each score, starting from 1970-1-1. Each user has at least 20 records. Movielens has three datasets of different sizes to adapt to different-sized algorithms. The smallest one has 100,000 pieces of data, while the biggest one has 1 million pieces.

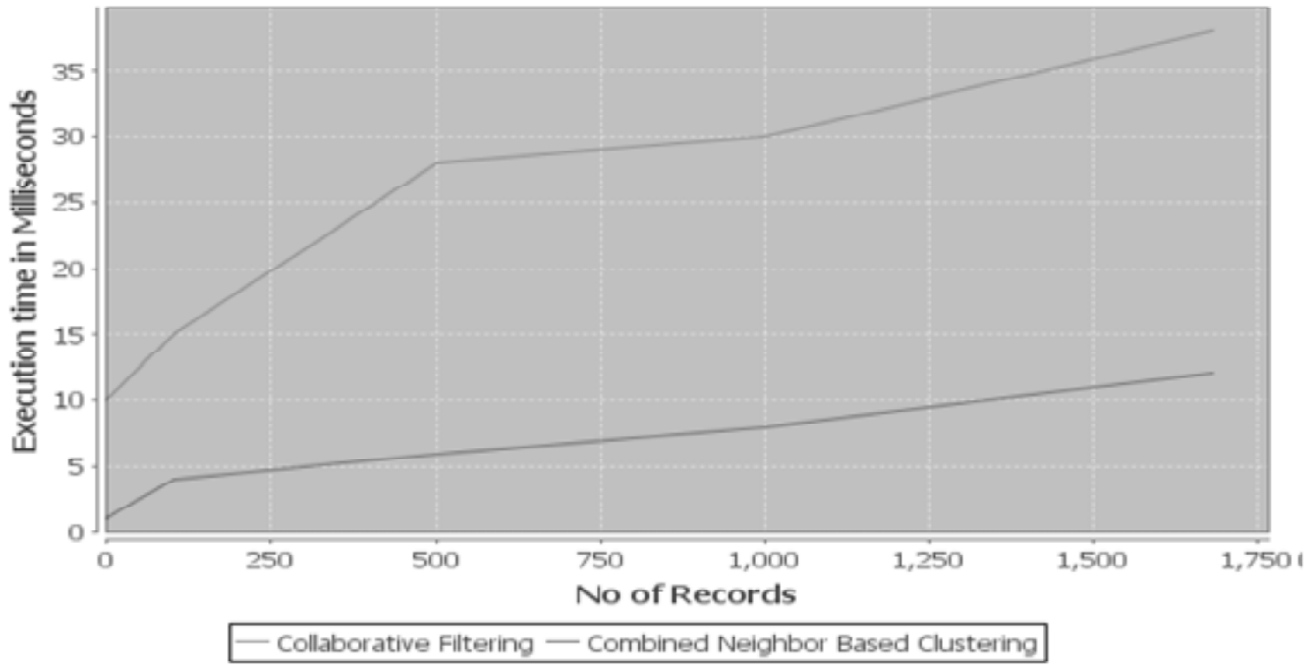


Figure 3: Traditional CF Vs. Proposed Hybrid CF

Small-scale dataset is used in the experiment. The dataset is analyzed and found that ratio of the numbers of users to numbers of items is at 56 and the sparsity is 99.36.

6.2. Evaluation Criteria

In this experiment we evaluated our algorithm on movielens dataset. Number of records is the number of movies and time of execution is calculated with respect to milliseconds, traditional collaborative filtering is compared with combined collaborative filtering. The time for this is linear as the underlying apriori based approach is expected to scale linearly with respect to number of records. Figure 3, shows how the relative execution time increases as we increase the number of records.

7. CONCLUSION

The hybrid collaborative filtering recommender system is built using the combined result of user based and item based collaborative filtering techniques. This allows the system to provide better accuracy and high quality in making recommendations for the products or items to the users. The future work is to improve the scalability of the collaborative filtering algorithm. Scalability can be improved by using Apache Hadoop and Mahout, which can handle very large data sets.

References

[1] Prem Melville, Vikas Sindhwani, Recommender System, January 2010.

[2] J. S. Breese, D. Heckerman, and C. Kadie, Empirical Analysis of Predictive Algorithms for Collaborative Filtering, in Proc. 14th Conf.Uncertainty in Artificial Intelligence, pp. 43-52, 1998.

[3] Dhoha Almazro, Ghadeer Shahatah, Lamia Albdulkarim, Mona Kherees, Romy Martinez and William Nzoukou, A Survey Paper on Recommender Systems, in Proc. of ACM SAC, pp. 1-11, 2010.

[4] Z. Zheng, H. Ma, M.R. Lyu, and I. King, WSRec: A Collaborative Filtering Based Web Service Recommendation System, in Proc. 7th International Conference on Web Services, USA, pp. 437-444, 2009.

[5] L.H. Ungar and D.P. Foster, Clustering Methods for Collaborative Fitering, in Proc. AAAI Workshop Recommendation System, pp. 285-295, 1998.

[6] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, Item-Based Collaborative Filtering Recommendation Algorithms, in Proc. 10th International Conference on World Wide Web, pp. 285-295, 2001.

[7] Manos Papagelis and Dimitris Plexousakis, Qualitative Analysis of User-Based and Item-Based Prediction Algorithm for Recommendation Agents, in Proc. Progress in Artificial Intelligence pp. 781-789, 2005.

[8] Yueping Wu and Jianguo Zheng, A Collaborative Filtering Recommendation Algorithm Based on Improved Similarity Measure Method, in Proc. of Progress in informatics and computing, pp. 246-249, 2010.

[9] SongJie Gong, A Collaborative Filtering Recommendation Algorithm Based on User Clustering and Item Clustering, in Proc. Journal of Software, pp. 140-159, 2010.

[10] Simon Renaud-Deputter, Tengke Xiong, Shengrui Wang, Combining collaborative filtering and clustering for implicit recommender system, in Proc. 27th International Conference on Advanced Information Networking and Applications, pp. 257-269, 2013.

[11] Thangavel Senthil Kumar, Swati Pandey, Customization of Recommendation System Using Collaborative Filtering Algorithmon Cloud Using Mahout, in Proc. Advances in Intelligent Systems and Computing, pp. 1-10, 2014.

[12] Smita Krishna Patil, Yogita Deepak Mane, Kanchan Rufus Dabre, An Efficient Recommender System using Collaborative Filtering Methods with K-separability, in Proc. International Journal of Engineering Research and Applications, pp. 31-35, 2012.

[13] Alejandro Bellogín, Iván Cantador, Fernando Díez, Pablo Castells And Enrique Chavarriaga, An Empirical Comparison Of Social, Collaborative Filtering, And Hybrid Recommenders, in Proc. ACM Transactions on Intelligent Systems and Technology, pp. 12-32, 2013.