

Graph based Recommendation System in Social Networks

Honey Jindal

Department of Computer Science and Engineering
JIIT, India

Anjali

Department of Computer Science and Engineering
JIIT, India

ABSTRACT

Media content recommendation is a popular trend now days. Twitter, Facebook, and Google+ are very popular in the world. The growth of social networks has made recommendation systems one of the intensively studied research area in the last decades. Recommendation systems can be based on content filtering, collaborative filtering or both. In this paper, we propose a novel approach for media content recommendation based on collaborative filtering. Firstly the user-user social network is created using most prominent neighbor set of each user by utilizing their preference information. Then these users are clustered using their neighbor sets and the user with maximum neighbor count is chosen as cluster head. When new user searches for its cluster its similarity is calculated with all the cluster heads. The user gets recommendation based on the average ratings of his cluster members. The proposed approach is tested on the users of Movielens Dataset. The proposed approach gives a hit ratio of 89.33%, Mean Absolute Error as 0.4756 and Root Mean Square Error as 0.7671 on Movielens dataset.

Keywords

Recommendation, social networks, content filtering, collaborative filtering, clustering, preferences, neighbor set.

1. INTRODUCTION

Recommendation system plays an important role in our lives. Accurate recommendations may help user to quickly identify their desirable items. It is widely used in online commercial sites to satisfy user's personal demands on the basis of their purchase behavior. Recommendation systems automatically recommends items to the target users based on their past purchases and behavior [1]. For example, a customer usually expresses an interest in an item either by viewing a product description or by placing the item in his "Shopping cart". Thus the customer will likely receive recommendations for additional products. These products can be recommended based on the top overall sellers on a site or on an analysis of the past buying behavior of the customer as a prediction of future buying behavior. Items are recommended to users based on their past ratings.

Recommendation systems can be summarized in three techniques: Content based filtering, collaborative filtering and Hybrid filtering technique. Content based filtering recommends resources to user according to past purchase history of the user. A major disadvantage of this method is that it uses only static information and the user interaction information is not fully utilized. Collaborative filtering relies on the past preferences or ratings correlation with other users. Based on this correlation, people with similar preferences are taken into account for recommendation. Hybrid methods are the combination of both content based and collaborative filtering.

The popular online social networking websites such as Facebook, twitter and YouTube provide novel ways for people to communicate and build virtual communities. The online shopping websites such as Amazon [1] [3], Flipkart and Snapdeal provides recommendation based on the purchase history and recently viewed items. This paper proposes a novel approach for media content recommendation in social network based on collaborative filtering. Firstly, a similarity matrix is calculated based on Euclidian distance between each user pair. Afterwards, a social network is formed where node represents the user and edges represents connection among user and the prominent neighbors with high similarity score. Then the clusters are formed based on the social network by k-means clustering algorithm. The node with highest closeness centrality is selected as the cluster head. When a new user arrives, its preferences are compared with the preferences of all cluster heads. The most similar cluster head is allotted to the new user. Thus the new user belongs to the most similar cluster and the average ratings of the cluster are recommended to the new user. This approach reduces the online computation time in cluster determination for the target user.

The remainder of this paper is organized as follows. First, we summarized the related work in section 2. Section 3 illustrates the proposed method in detail. Section 4 includes complexity analysis. Section 5 shows the result and simulation of proposed method. Section 6 gives discussion, followed by conclusion and future directions for research in section 7.

2. RELATED WORK

In this section, some of the prior techniques for recommending resources to the users are briefly reviewed. A recommender system is an effective tool used to reduce information overload while searching content, product information or documents on the internet. The recommendation problem occurs each time as user enters in the system. Consider a recommendation system with M users and N items shown in Fig.1. The relationship between user and item is represented by $M \times N$ rating matrix, R . Each entry in R_{ij} represents rating given by the user i to item j which scales from 1 to 5. An item with no rating represents as $R_{ij}=0$.

Recommendation algorithm is the core of every recommendation system, which determines the performance of the system. There are three popular methods adopted by most of the recommendation systems. Firstly, the content-based recommendation method [4] [7] [8], which recommends resources based on the past purchase history and not on user's preferences and opinion. One of the disadvantages of this method is that each resource is defined by their associated features and resources are recommended to the target user based on similarity of the resource features not by utilizing the taste and preferences of the target user.

User/Item	Item 1	Item 2	Item 3	Item 4	Item 5
User1	5	0	0	3	0
User2	0	4	3	0	2
User3	2	0	5	1	0
User4	1	3	0	0	3

Figure 1: User-Item Rating Matrix

Secondly, the collaborative filtering recommendation method [2], which recommends resources based on user's previous ratings and rating of same items by the group of similar user's. It is based on the assumption that similar group of user express similar interest in similar resources. These collaborative filtering systems can be classified into user-based and item-based systems. User-based systems use cosine-based similarity or Pearson correlation to find a group of similar users which is identified by setting some threshold or selecting top-N similar users. Item-based systems predict similar group of items and then takes a weighted combination of their ratings to identify the list of users for whom they get recommended. This method improves quality of recommendation by using user past preferences but completely ignores content information.

Thirdly, social network recommendation method [2][10], which uses some techniques of graph theory to find strongly connected components - graph cut, bridges, minimum spanning tree, breadth first search, depth first search, clique, modularity maximization, etc. These graph based recommendations use transitive association between users and resources in the bipartite user-resource graph. Such recommendation problem turns into node selection problem on a graph and requires more computation cost in online community detection.

Several techniques have been proposed to address recommendation problem like KNN, clustering, customer profile, dynamic network reformation [4], event driven recommendation, category correlation [7], genre correlation and recent profile-based filtering etc. In real time scenario updates of customers profile upon transactions and reformations of the network are too burdensome due to heavy amount of transactions whereas K-Nearest Neighbor (KNN) algorithm utilizes an entire database of user preferences to compute recommendations. This algorithm tends to be simple to implement and require no offline cost. But as the size of user and item increases, the online performance of this algorithm tends to decrease. KNN takes $O(M*N)$ time in online mode to recommend items, which is very high.

The proposed method first calculates the similarity among users and forms an edge between most similar users. Afterwards, users are clustered using their neighbor set and the cluster head is identified from each cluster that has maximum neighbor count. This method assumes that cluster head is the main representative of the cluster who has maximum closeness centrality among other cluster members. Thus, while searching similar community only cluster head preferences need to be compared. Once community is detected, a list of non-zero average rated items is recommended to the user which is rated by the community

members of the identified community. In the proposed method some irrelevant links were discarded from the start of offline mode and recommendation computation is calculated only among the identified community members. Thus the proposed method reduces computation cost and online time complexity.

3. PROPOSED METHOD

In order to recommend items to the user, the proposed method uses the combination of collaborative filtering and social networking recommendation system. Overall architecture of the proposed method is shown in the fig.2.

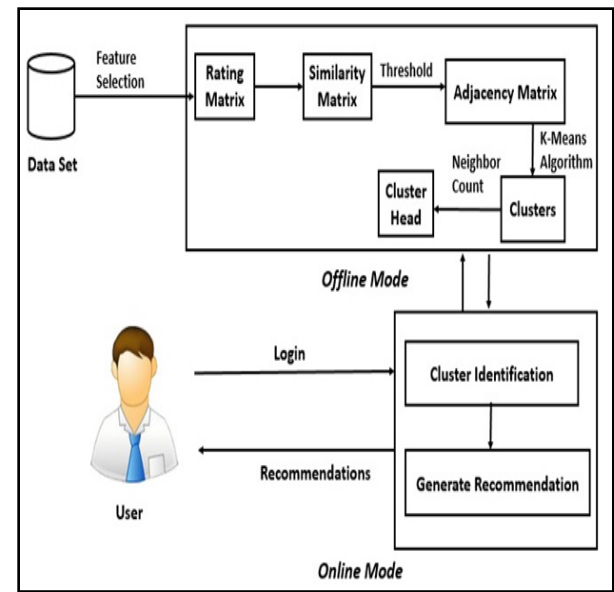


Figure 2: Overall Architecture of Proposed Method

The algorithm works in two modes: *Offline mode* and *online mode* described as follows-

A. Offline mode

1. Construct a rating matrix, R_{ij} from given dataset and initialize non-rating items as 0 as shown in Fig. 1.
2. Calculate the similarity matrix, S_{ij} among each user pair using Euclidean distance.
3. Construct an adjacency matrix, A_{ij} based on threshold, Θ .

$$A_{ij} = \begin{cases} 0 & \text{if } s_{ij} > \Theta \\ 1 & \text{if } s_{ij} \leq \Theta \end{cases}$$

4. Apply K-means algorithm to cluster similar users.
5. Calculate neighbor count corresponding to each user, N_i where i represents the user.
6. For each cluster, calculate cluster head with maximum neighbor count.

$$CH_k = \max(N_i) \quad \forall i \in k.$$

B. Online mode

When new user logs in the system-

1. Extract user's preference information.

2. Identify the best cluster head by comparing preferences of target user with each cluster head.
3. Identify the users in the cluster of the matching cluster head.
4. Identify the movies rated by the users in the particular cluster.
5. Take average of the identified movie ratings.
6. Display the list of movies to the user based on average ratings given by the matching cluster members.

4. COMPLEXITY ANALYSIS

In offline phase, described in section 3(A), a rating matrix is formed with $M \times N$ dimensions where M is the total number of users in training set and N is the total number of items. The rating matrix formation takes $O(M \times N)$ time. A similarity matrix is constructed based on user's preferences which requires $O(M^2 N)$ time. An adjacency matrix is generated using some threshold value which requires $O(M \times M)$ complexity. Afterwards, K users are chosen as centroid to cluster the users into K similar groups using K-Means algorithm. 3(A(4)) requires $O(KM^2)$ time. Since the value of K is very less as compared to the number of users and items, it could be ignored. Thus, total time taken to the user clusters is $O(KM^2) \approx O(M^2)$. In 3(A(5)), number of neighbors corresponding to each user in the cluster is determined which requires $O(KM'M)$ time where M' is the number of users in the cluster. Since K and M' is very less as compared to M , they could be ignored. Thus, the total time required to count the number of neighbors is $O(M)$. In 3(A(6)), cluster head is determined from each cluster. A cluster head is the user which is most popular among its cluster members. Thus, a node with maximum neighbor count is considered as cluster head which requires $O(KM')$ time. Time required in offline mode is $O(MN) + O(M^2 N) + O(M^2) + O(M^2) + O(M) + O(KM')$. Since, $M^2 N > MN > M^2$ the offline time complexity is $O(M^2 N)$.

In the online phase, described in 3(B), the maximum time required to determine the matching cluster is equivalent to the total number of cluster heads. Since there is only one cluster head for each cluster, the number of cluster heads is equal to the number of clusters i.e., K . Therefore time required to determine the matching cluster is $O(K)$. Calculating the average rating will take $O(M'N)$ time where M' is the number of users in the cluster that has been identified for the target user. Since $M' < N$ prediction for the ratings takes $O(N)$ time. Thus, the total time required during online phase is $O(N)$.

5. EXPERIMENTAL RESULTS

5.1 Experimental Setup

This study uses Movielens dataset with 100,000 ratings, 1682 movies and 943 users who rated movies from the degree 1 to 5. This dataset is taken from Grouplens (<http://grouplens.org>). The training set is divided in the ratio of 0.8 whereas testing set is divided in the ratio of 0.2.

5.2 Evaluation Metrics

Three statistical metrics are used to evaluate the accuracy of our method to compare the predicted values with the user-provided values. More precisely, three popular metrics are used: Hit Ratio metric (Eq 1), Mean Absolute Error (MAE) metric (Eq 2), Root Mean Square Error (RMSE) metric (Eq 3).

$$\text{Hit Ratio} = \frac{\text{Number of matches with desired output}}{\text{Total number of recommended items}} \quad \text{Eq(1)}$$

$$\text{MAE} = \frac{\sum_{i \in D} |p_i - r_i|}{|D|} \quad \text{Eq (2)}$$

$$\text{RMSE} = \sqrt{\frac{\sum_{i \in D} (p_i - r_i)^2}{|D|}} \quad \text{Eq (3)}$$

Here, p_i is the predicted rating, r_i is the real rating. D is the test data collection. For a high quality recommendation system low MAE, RMSE and high hit ratio is derived.

5.3 Results

Following are the results of three measures to check the efficiency of the system.

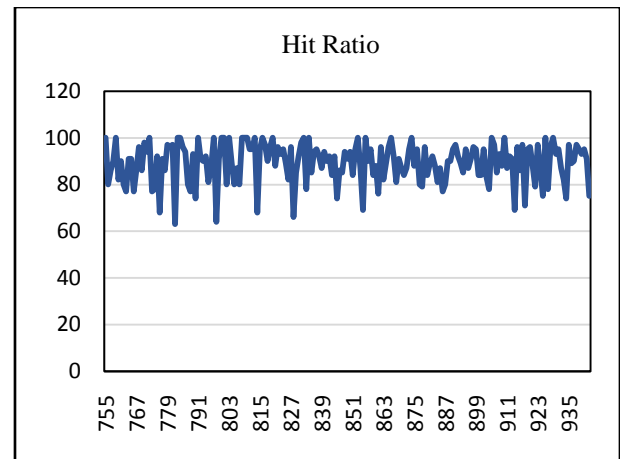


Figure 3: Hit Ratio of the test set users

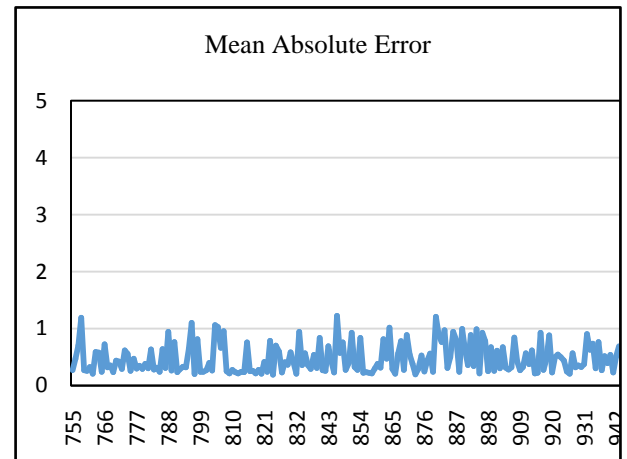


Figure 4: Mean Absolute Error of the test set users

Fig.3 shows the hit ratio of test set users where x-axis denotes the user-id and y-axis denotes the hit ratio for the recommended movies. Hit Ratio denotes the percentage of correctly recommended movies. It is clearly visible that most of the user have hit ratio in between 80 to 100. Higher the hit ratio more accurately the system is able to recommend item to the user.

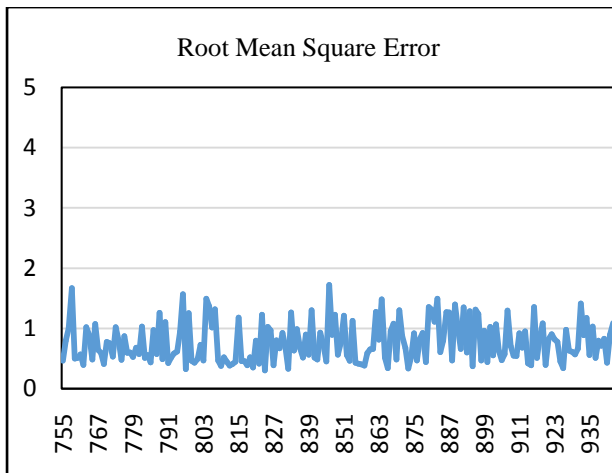


Figure 5: Root Mean Square Error of test set users

Fig. 4 shows MAE for each test set user. The MAE for the recommended movies lies in between 0.2 to 1.2. Most of the users have MAE in between 0.2 to 0.6. Fig. 5 shows the RMSE for each test set user. There is large variation in the RMSE value for recommended movies which ranges from 0.3 to 1.7. The figures show that most of the users have high hit ratio and low MAE and RMSE value.

6. DISCUSSION

The experimental results show that the proposed method has huge improvement in terms of MAE and RMSE values shown in Table 2. In proposed approach both MAE and RMSE value is low as compared to previous work which is necessary for every recommendation system. The similarity between real rating and predicted rating is 89.33%. Table 3 shows the comparison of proposed method with existing methods.

Table 1: Accuracy measures for recommended movies

Measures	Accuracy for Recommended movies
Hit Ratio	89.338
MAE	0.4756
RMSE	0.7671

KNN and User-based collaborative filtering requires high online cost as compared to community based user domain model collaborative filtering algorithm (CUCRA) [2] and proposed method because in both KNN and user based CF method entire dataset needs to be considered in order to find similar neighbor set whereas CUCRA and proposed method forms cluster in offline mode. Thus there is no need to search the entire dataset to determine the best cluster. The online cost of proposed method is equivalent to community based method.

Table 2: Comparison of rating prediction quality from existing methods

Methods	MAE	RMSE
KNN	0.7296	0.9679

User-based CF	0.6861	0.9074
Community based	0.6919	0.9602
Proposed Method	0.4756	0.7671

Table 3: Comparison of online time complexity with some existing methods

Methods	Time Complexity (Online)
KNN	$O(mn)$
User-based CF	$O(mn)$
Community based	$O(n)$
Proposed Method	$O(n)$

7. CONCLUSION AND FUTURE WORK

This paper presents a novel approach for media content recommendation based on collaborative filtering to recommend resources close to the user preferences. This approach works in two modes- offline and online mode. In offline mode, the user-user social network is created using most prominent neighbor set of each user by utilizing their preference information. Then these users are clustered using their neighbor sets and the user with maximum neighbor count is chosen as cluster head. In online mode, when new user searches for its cluster its similarity is calculated with all the cluster heads. The user gets recommendation based on the average ratings of his cluster members.

The experiment shows that the proposed method achieves the improvement in terms of MAE, RMSE and has advantages over existing solutions. Time required during the online phase (discussed in section 4) shows a significant improvement over the existing methods (shown in table 2). Recommending a list of items to the new user takes less time as compared to prior methods (shown in table 3).

In the future, we will evaluate our approach by comparing it with other related methods (such as machine learning algorithms, statistical learning theory, etc) through simulation experiments. Although we achieve significant improvement over accuracy and reduces the search time required in online mode but search time required in offline mode also needs improvement. We would further extend our approach into distributed environment with large real time dataset.

8. REFERENCES

- [1] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions," IEEE Trans. Knowl. Data Eng., vol. 17, no. 6, pp. 734-749, Jun. 2005.
- [2] FulanQain, Yanping Zhang, Yuan Zhang, and Zhen Duan, "Community-based user domain model collaborative recommendation algorithm," TSINGHUA Science and technology, ISSNII 1007-0214/1103/1011, vol. 18, no. 4, pp. 353-359, August 2013.
- [3] Lalita Sharma, Anju Gera, "A Survey of Recommendation System: Research Challenges," International Journal of Engineering Trends and Technology (IJETT), vol. 4 Issue 5, May 2013.

- [4] Hyeakyeong Kim, Young U. Ryu, Yoonho Cho, and Jae Kyeong Kim, "Customer-Driven Content Recommendation Over a Network of Customers," *IEEE transactions on system, man and cybernetics-part a: systems and humans*, vol. 42, no. 1, January 2012.
- [5] Michael Steinbach, George Karypis, Vipin Kumar, "A Comparison of Document Clustering Techniques," Department of Computer Science and Engineering, University of Minnesota.
- [6] Liang Hu, Wenbo Wang, Feng Wang, Xiaolu Zhang, Kuo Zhao, "The Design and Implementation of composite Collaborative Filtering Algorithm for Personalized Recommendation," *Journal of software*, vol.7, no. 9, 2012.
- [7] Sang-Min Choi, Yo-Sub Han, "A Content Recommendation System Based on Category Correlations," *Fifth International Multi-Conference on Computing in the Global Information Technology*, 2010.
- [8] Mark Claypool, Anuja Ghokale, Tim Miranda, Pavel Murnikov, Dmitry Netes, Mathew Sartin, "Combining Content-Based and Collaborative Filtering in an Online Newspaper," *ACM SIGIR Workshop on Recommendation System implementation and Evaluation*, 1999.
- [9] Jonathan L.Herlocker, Joseph A. Konstan, Loren G.Terveen, John T.Reid, "Evaluating Collaborative Filtering Recommender Systems," *ACM Transactions of Information Systems*, vol. 22, No.1, January 2004.
- [10] Ziqi Wang, Yuwei Tan, Ming Zhang, "Graph-Based Recommendation on Social Networks," *12th International Asia-Pacific Web Conference*, 2010.
- [11] Lubos Demovic, Eduard Fritscher, Jakub kriz, Ondrej Kuzmik, Ondrej Proksa, Diana Vandlikova, Dusan Zelenik, Maria Bielikova, "Movie Recommendation Based on Graph Traversal Algorithms, " *24th International Workshop on Database and Expert Systems Applications*, 2013.
- [12] Ioannis Konstas, Vassilios Stathopoulos, Joemon M Jose, "Social Networks and Collaborative Recommendation," *ACM SIGIR*, 2009.
- [13] Jun Li, Shuchao Ma, Shuang Hong, "Recommendation on Social Network Based on Graph Model," *proceedings of the 31st Chinese Control Conference*, 2012.