I.J. Information Technology and Computer Science, **2016, 7, 72-80**
Published Online July 2016 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijitcs.2016.07.10

Pre-Recommendation Clustering and Review Based Approach for Collaborative Filtering Based Movie Recommendation

Saudagar L. Jadhav
Department of Computer Engineering, VIIT, Savitribai Phule Pune University, Maharashtra, India
E-mail: saudagarljadhav@gmail.com

Mrs. Manisha P. Mali
Department of Computer Engineering, VIIT, Savitribai Phule Pune University, Maharashtra, India
E-mail: manisha.mali@viit.ac.in

***Abstract*—**The recommendation is playing an essential part in our lives. Precise recommendations facilitate users to swiftly locate desirable items without being inundated by irrelevant information. In the last few years, the amount of customers, products and online information has raised speedily and results out into the huge data analysis problem for recommender systems. While handling and evaluating such large-scale data, usual service recommender systems regularly undergo scalability and inefficiency problems. Nowadays, in multimedia platform such as movie, music, games, the use of Recommender System is increased. Collaborative Filtering is a dominant filtering technique used by many RSs. CF utilizes the rating history of the user to find out "like minded" users and this set of like-minded user is then used to recommend the movies which are liked by these like-minded users but did not watch by the active user. Thus, in CF, to find out the "neighborhood" the rating history of a user is used, but the reason behind the rating is not considered at all. This will lead to inaccuracy in finding a neighborhood set and subsequently in recommendation also. To cope with these scalability and accuracy challenges, this paper proposes an innovative solution, Clustering and Review based Approach for Collaborative Filtering based Recommendation. This innovative approach is enacted with the two stages; in the first stage the clustering of the available movies for recommendation is clustered into the subclasses for further computation. In the succeeding stage, the methodology based on reviews is utilized for finding neighborhood set in User Based Collaborative Filtering.

***Index Terms*—**Recommendation Systems, Collaborative Filtering, Clustering, Accuracy, Review.

I. INTRODUCTION

The popularity of Internet made the rapid increase in e-commerce as well as multimedia data and due to this, users are pullulated by choices to chew over and that they might not have the time to personally assess these choices. In this era of information burden it is not easy to achieve what users want, because users sometimes do not know what they want to search for. Recommender Systems (RSs) play a dominant role in discovering valuable and interesting information for users searching among massively large databases. A RS helps users that don't have any comfortable competency to judge the, probably overwhelming, variety of alternatives. In their simplest type RSs gives a customized and stratified list of things by predicting what the foremost appropriate items are, supported the user's history, preferences and constraints [1, 2, 3]. Nowadays, in multimedia platform such as movies, music and games the RS is widely used. Different Movie Recommendation systems are helping users by recommending different movies to them.

These RSs are mainly based on user profile and can be divided in four categories based on how user profile information is used: Demographic Information Filtering, Content Based Filtering, Collaborative Filtering (CF) and Hybrid Filtering.

A. Demographic Filtering

Demographic information means the personal attributes such as age, gender, occupation, nationality which describes the individuals. This Demographic information may be used to categorize the user on the basis of common personal attributes. Demographic filtering is relying on the principle that persons having common demographic information will also have common preferences. The main advantage of this filtering is that, it does not rely on user rating history because it completely supports Demographic data of user [4, 5].

B. Content Based Filtering

Content based filtering is totally depends on the historical data of users' choices. Content based approach recommends movies similar to movies which are previously preferred by user [6]. The Content based recommendation method primarily consists in matching up the attributes of the user profile against the attributes of a content object i.e. Item profile. The result is an impact judgment that denotes the user's level of interest in this object or the chance that the user goes to like that object.

C. *Collaborative Filtering*

The CF method is based on the principle that if two users have same or almost same commonly rated items, then they may have similar preferences or tastes. Such users are called as similar users. The basic aspiration of CF method is to build the user-item matrix by collecting user preferences or activities and try to find out the users with same interest. The users which have the analogous interest bring into the group called as a neighborhood. The items which are unrated by the user but rated by his neighborhood are recommended to this user [8].

D. *Hybrid Filtering*

For efficient and accurate results, many RSs combines different filtering techniques with each other. In this Hybrid method the combinations like CF with Demographic Filtering, Collaborative Filtering with Content-based Filtering can be used. In another way we can incorporate probabilistic methods such as Clustering, Decision trees into the Collaborative Filtering [7]. These combinations of different approaches proceed in a different manner to achieve the desired goal.

II. MOTIVATION

The RSs are the foremost dominant and therefore the promising technology whereas creating selections. However, today the explosive growth of e-commerce and online environments have created the difficulty of data, search and choice progressively serious; Users are full of choices to contemplate and that they might not have the time or knowledge to appraise these choices in person [1, 2]. This overloaded information creates a problem for RSs also. RSs suffers the problem of analyzing these data in a timely manner. The performance of traditional CF based RSs is getting reduced when the data is in vast amount and the data is changing dynamically.

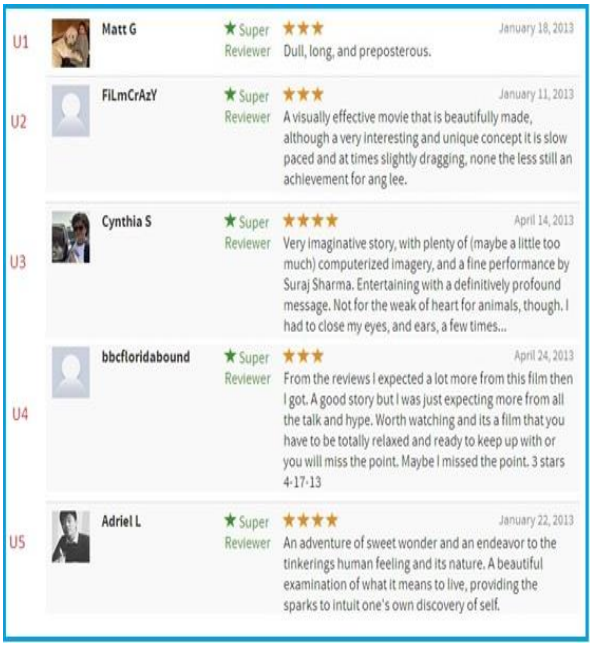
Most of CF based RSs found to be utilizing the rating oriented data for making Recommendations. While making the recommendations to any active user, RS first finds out few users which have same likeliness as an active user. This similarity is found out on the basis of the rating given by active user as well as other users to a particular item say movie. After finding out the similar users, RS recommends movies which are unrated by the active user but rated by his neighbor. Thus CF based RS leads to the problem of inaccuracy by considering the rating as parameter to find similar users. It is found that the Recommendations made by these RSs are not that much ideal or accurate for the active user. Sometime the recommendations given are inappropriate to the active user. Following example will depict this scenario more precisely.

Example: Let’s consider the following rating and review example from RottenTomatoes site to Life of Pie movie, which will illustrate the accuracy problem in the rating oriented approach:

Table 1. User Movie Rating Matrix

	U1	U2	U3	U4	U5
Life of pie	***	***	****	***	****
M2					
M3	****				
M4				****	

In Fig. 1 and the Table 1, Users U_1, U_2, U_4 gives three stars to the movie “Life of Pie” and users U_3, U_5 gives four stars to the movie “Life of Pie”. So, according to the rating we consider that users U_1, U_2 and U_4 has the same likeliness as they gave same rating three to the same movie. Similarly the U_3 and U_5 are treated as similar. Now suppose user U_2 is the active user. We found out that U_1, U_2 and U_4 has same likeliness so the existing recommendation system recommend other movies that is M_3 and M_4 to the user U_2 as M_3 is liked by U_1 and M_4 by U_4. But are the users U_1, U_2 and U_4 have the same likeliness?? The answer is probably no, because different users may like different features of different movies. Sometimes, users may rate movies similarly, but their ratings may be based on different features of the movie. Thus the RSs based on these ratings, which did not consider the reason behind the rating given by user i.e. Feature about that movie may lead to inaccurate recommendations.



Matt G ★ Super Reviewer ★★★ January 18, 2013
Dull, long, and preposterous.

FiLmCrAzY ★ Super Reviewer ★★★ January 11, 2013
A visually effective movie that is beautifully made, although a very interesting and unique concept it is slow paced and at times slightly dragging, none the less still an achievement for ang lee.

Cynthia S ★ Super Reviewer ★★★★ April 14, 2013
Very imaginative story, with plenty of (maybe a little too much) computerized imagery, and a fine performance by Suraj Sharma. Entertaining with a definitively profound message. Not for the weak of heart for animals, though. I had to close my eyes, and ears, a few times...

bbcfloridabound ★ Super Reviewer ★★★ April 24, 2013
From the reviews I expected a lot more from this film then I got. A good story but I was just expecting more from all the talk and hype. Worth watching and its a film that you have to be totally relaxed and ready to keep up with or you will miss the point. Maybe I missed the point. 3 stars 4-17-13

Adriel L ★ Super Reviewer ★★★★ January 22, 2013
An adventure of sweet wonder and an endeavor to the tinkerings human feeling and its nature. A beautiful examination of what it means to live, providing the sparks to intuit one's own discovery of self.

Fig.1. User Reviews from Rotten Tomatoes.
(Source:http://www.rottentomatoes.com/m/life-ofpi/reviews/?type=user)

Now concentrating on the user’s reviews to the movie Life of Pie in Fig. 1, we can say that the user U_2 and U_3 have same likeliness as they both like the visual effects given to the movie. These features of movies like story, acting, actor effects and so on can be obtained from the users reviews only. So considering these features or reviews to find out the similar users or users have same likeliness will give more accurate recommendation to the active user. As these features are obtained only from the reviews given by user to a particular movie, we have to move from Rating based approach to Review based approach for computing the similar users.

There are few Review based RSs exists, but many of these undergo with the scalability problem. These Review based RSs did not consider clustering or any preprocessing of reviews for the further computation. If the system did not carry out the clustering of items say movies, then we have to consider almost all the Movies and their respective reviews for further computation, i.e. Similarity Measurement and Recommendation. So this Computation may take more time than Clustering based computation and subsequently undergo with scalability problem.

From the above example, it is clear that, the current RSs are facing the problems like scalability and inaccuracy. The main challenges which faced by the RSs are: (1) To make the decisions within acceptable time. (2) To make the recommendations ideal for users.

Motivating by above challenges, in this paper, we address these aforementioned challenges by considering following key motivating factors: (1) To reduce the number of movies required for computation. (2) To make use of reviews to find the neighborhood set required for recommendations.

III. Related Work

The Scalable Data having sparsity, results a poor quality performance. To deal with this problem, a good solution is proposed in [9, 12, 13]. The solution is based on clustering the data based on the rating given by the users to different items. In this the users are clustered on their ratings given to items. In [9], the similitude threshold model is used to reduce the vigorously changing item space into clusters. The similarity computation is carried out between the current item and the center of the cluster. In [12], for every new user his rating is compared to the cluster centers and according to it, that the new user is clustered. Then the item cluster is created in the same manner and used for recommending the items to particular user cluster. In [13] same clustering strategy is applied. These methods have proven to be good for scalable data having Sparsity. To deal with scalability issues, these methods utilize the cluster of similar user/items to the target user/item and all further computation is performed on this cluster only. The MCT i.e. Mean Consumed Time of these approaches is found lower than other existing approaches. But in these methods, it is possible that the too many items/users can involve in a single cluster. The MAE of these methods is found higher than another item based CF.

In CF the similarity computation between every pair of services or users is a crucial and time consuming step and may consider many services irrelevant to the user. It oversteps the processing capability of RSs and may affect the accuracy of predicted rating. To attack these problems in [10, 11, 15], a new method is proposed for hierarchical clustering of data as well as user. In [10], Items are clustered on the basis of Mean Squared Distance between items and clusters. The Sparsity of data is minimized by replacing the unknown values with the center value of parent cluster for a particular item. This method found useful for new user as well as for sparse data also. In [11], the users are clustered depending upon their ratings to the item by using Top-down divisive clustering approach. This approach found very useful for solving the scalability problem when data size is too large. The accuracy achieved in both these approaches depends on the neighborhood size. Similarly, in [15], firstly all the services are recruited into some clusters based on their similarities using AHC algorithm and then the CF is applied within a cluster to compute the rating similarity and recommend ideal services to the user. This reduces the time required for CF to compute rating similarity significantly and also enhance the accuracy of RSs. The top-down clustering of data and user are carried out independently based on ratings given by user and rating of items.

Grouping of items or users gives accurate recommendation and help to reduce the Sparsity of data. The [14] proposed a statistical model for CF which helps to handle the clustering considering various properties of items or users under consideration. In this, the users and corresponding items individually divided into the clusters and there is a probability link between the user cluster and item cluster. Gibbs sampling used for this method is working well, but the cost of computation is somewhat high.

The limitations of Traditional similarity measures such as PCC and Cosine as well as the Cold start problem are addressed in the [16].This paper come up with a novel Similarity measures called PIP measure. PIP utilizes only the domain specific meaning of user rating. PIP has better performance for users those leads to the Cold-start problem.

If the preferences changing with time are not considered for the Recommendation, then it will lead to incorrect recommendations. In [17], the new CF method which considers the users changing interest with the time is put-forth for accurate recommendation results. The similar items are gathered together by Clustering and then for each item in the cluster, the user preferences are calculated by previous given preferences on item in the cluster and the corresponding time of preference to each item also. The consideration of changing interest of users will lead to the reliable selection of neighborhood and better performance over existing CF. This method needs the setting of parameters such as the number of clusters, number of neighborhoods and the threshold for recent time to the particular values only.

To solve the problems of new users, the paper [18] proposes a solution based on creating a similarity network of reviewers preferences. From the reviews of products given by reviewers, the reviewer’s weights on their preferences are calculated and then the network of similar preferences is created. The sub network of the similar users of this network is identified by using Latent Class Regression Model (LCRM). The similarity computation is carried between active and other user's preferences within the relevant sub-network only.

Most of the users do not rate the enough hotels or products and this will lead to cold start problem for RS. To overcome this, the paper [19] proposes a solution based on the text of the reviews from various hotel reviewers. The texts from the reviews are mined and the analysis is carried out for a common group having common context. Common group means the purpose of the trip, the nationality of the user and the context group means the locations, service or food or any hotel related parameters. The trip purpose, nationality and the required hotel context are taken from the active user and similarities are measured with the mined text from reviews. The most similar reviewers are found out and the most preferred hotels by them are then recommended to the user.

Current RSs works on a particular score given by a user to a product, instead of taking into account the particular reason behind assigning the score to that product and may lead to inaccurate recommendations. This paper [20] proposed a method which considers the user reviews to calculate the user similarities. This paper utilizes the Latent Semantic Analysis (LSA) for the similarity computation.

The existing RSs recommend services or products to the different users based on same rating and ranking of services. It did not consider any specific user preferences and hence it is not much useful for personalized recommendation. To address the above challenges, the paper [21] proposed a Keyword- Aware Service Recommendation (KASR) method. The different user preferences are indicated by the keyword set. This keyword set is used by a user based CF algorithm to generate an ideal recommendation to the user.

Thus we can say that the existing Recommendation systems are mainly either based on clustering and rating (Clustering and then Rating) of movies or only review based. So there is an accuracy problem with a rating based RSs. And RS based on only reviews faces the problem of scalability for similarity measurement and recommendations as whole item set is used for computation. To overcome these challenges the next section puts forth the novel solution.

IV. PROPOSED SYSTEM

To overcome the challenges mentioned in Section 2 and 3, we proposed a novel solution in this section. The main goal of the proposed system is to make a Recommendation System based on Clustering of the Movies that we are going to recommend, and then finding the neighborhood set for the active user by utilizing the reviews of different movies within the already formed clusters only. The clustering is carried out for the purpose of reducing the item space on which the further computation of similarity measurement and recommendation is based.

The Proposed RS in this paper has two main phases:

A. Clustering of Movies
B. Review Based CF

A. Clustering of Movies:

It is basically categorizes the available movies into different subclasses based on their features such as Actor, Actress, Writer, Director, Genre. It has the following sub-modules:

1) Compute Feature Similarity:

Different movies have different features such as Actor, Actress, Writer, Director, Genres. In this paper the similarity between movies on the basis of their features is carried out by using Jaccard Similarity Coefficient (JSC). Since in the Movie Feature Sets, the distance is determined by how many different and how many same features are there in the sets, we decided to use the JSC for similarity computation. JSC is the statistical measure of similarity between different sample sets of movie features. For two Feature sets, JSC is defined as the cardinality of their intersection divided by the cardinality of their union. Let F_1 and F_2 be the Feature set of two different Movies M_1 and M_2 respectively. The Feature Similarity between movie M_1 and M_2 is computed by following equation (1):

$$F_{sim(M_1,M_2)} = \frac{|F_1 \cap F_2|}{|F_1 \cup F_2|} \tag{1}$$

From equation (1) we can say that if the value of $|F_1 \cap F_2|$ is larger, and then the similarity is more between two movies. Denominator $|F_1 \cup F_2|$ is works as a scaling factor for ensuring that $F_{sim(M_1,M_2)}$ is in between 0 to 1.

2) Cluster the Movies:

Clustering is the important step in the proposed approach as it reduces the available Movie dataset for further computation as well as the movies between each cluster are more similar to each other than movies in another cluster. In this phase the available movies are clustered on the basis of their feature similarity.

Generally the clustering is carried out when there is a huge amount of data. There are different clustering techniques involved, which follow either hierarchical or partitional approach. Partitional approach such as k-means clustering has some limitations such as we to have to give the number of clusters k at the start of algorithm, premature termination of the algorithm. Hierarchical clustering is generally the family of clustering algorithms that builds nested clusters by merging them successively. This hierarchy of clusters is represented as a tree. The root of the tree is the unique cluster that gathers all samples; the leaves represent only one sample. The hierarchical clustering methods are further classified into

Copyright © 2016 MECS

agglomerative or divisive on the basis of formation of clustering hierarchy either in bottom-up or top-down fashion.

Due to the simple processing structure and the better performance of Agglomerative Hierarchical Clustering Algorithm (AHC), we have decided to use it in our proposed RSs to cluster the movies. The beauty of this AHC is that we don't have to provide the size of cluster in advance. Instead of that, we can initialize the algorithm with a large cut height of the hierarchy and decrease the height if we found most movies in one cluster. In this way, we can provide the variable cluster size to fit the user's interests. It is shown in Fig. 4.

Following is the AHC algorithm for clustering the movies. The input to the algorithm is the set of n movies, feature similarity matrix computed in the previous step and the number of clusters k. At the start, each movie is assigned to be a cluster of its own. The two most similar clusters are merged in reduction step. Reduction step is repeated, until only $k(k<n)$ clusters remain.

Algorithm: AHC algorithm for Movie clustering

```
Input:  A set of $n$ Movies $M = \{m_1, m_2, m_3, ......m_n\}$,
        A Feature similarity matrix $D = [d_{i,j}]_{n \times n}$,
        The number of required clusters $k$.
Output: $Dendrogram_k \ for \ k = 1 to |M|$.
1.  $C_i = \{m_i\}, \forall_i;$
2.  $d_{C_i,C_j} = d_{i,j}, \forall_{i,j};$
3.  $for \ k = |M| \ down \ to \ k$
4.      $Dendrogram_k = \{C_1, C_2, C_3, ......C_k\};$
5.      $l.m = argmax_{i,j} d_{C_i,C_j};$
6.      $C_l = Join(C_l, C_m);$
7.      $for \ each C_g \in M$
8.          $if C_g \neq C_l and C_g \neq C_m$
9.              $d_{C_l,C_g} = Average(d_{C_l,C_g}, d_{C_{lm}C_g});$
10.         $end \ if$
11.     $end \ for$
12.     $M = M - \{C_m\};$
13. $end \ for$
```

B. Review based CF:

This is the online phase. It is basically applied to user based collaborative filtering stage. In this phase the neighborhood set for the active user is found out and it is further utilized for recommending movies to the active user. This phase involves following sub-modules:

1) Cluster Selection:

This phase utilizes the clusters formed in the previous phase. The query is taken from the active user to find out the most appropriate and relevant cluster to the user from all available clusters. The similarity between user query and clusters is found out by using the JSC has given by the equation (1). The cluster having maximum similarity with user query is considered as an appropriate and relevant cluster to the user. The further computation such as finding a neighborhood set and recommendation is carried out within this selected cluster only. This will help to reduce the data set required for computation.

2) Review based User Similarity within Selected Cluster:

In this phase all the other users who have same likeliness with the active user is found out. This set of likeminded user is called as a neighborhood. This neighborhood set is found out using the reviews given by active user and other users to the movies in the particular cluster. The similarity between reviews given by active user and the reviews given by another user is computed using the JSC mentioned in following equation (2).

$$R_{Sim(U_a,U_j)} = \frac{|R_a \cap R_j|}{|R_a \cup R_j|} \qquad (2)$$

Where R_a is the review given by user U_a and R_j is the review given by U_j.

3) Select Neighbors:

In this the Top-K users from the similar user found in above step are selected as a neighbor of active user. Based on the Review similarities between different users, the neighborhood set of an active user U_a is formed by using following equation (3).

$$N(U_a) = \{U_a | R_{Sim(U_a,U_j)} > \alpha, U_a \neq U_j\} \qquad (3)$$

Here, $R_{Sim(U_a,U_j)}$ are the review similarities between active user U_a and other users U_j, Which is obtained by the formula (2). The α is set as a threshold value of review similarities, so that we can obtain the neighborhood set of active user U_a. The user U_j will be selected as neighbor of active user U_a and put it in neighborhood set $N(U_a)$ only if it satisfies the condition; $R_{Sim(U_a,U_j)} > \alpha$.

4) Compute Rating Prediction:

Once the set of most similar users i.e. Neighborhood set is found out, the personalized rating for each candidate movie is computed. At the end, the personalized movie recommendation list will be presented to the user and the movies, with the highest ratings will be recommended to the active user. In this step the personalized rating of candidate movies are calculated and the movies are ranked according to their ratings and presented as recommendations to the active user. The rating prediction is carried out using the following equation (4).

$$P_{u_a,m_i} = \overline{r_{u_a}} + \frac{\sum_{u_j \in N(U_a)} R_{Sim(U_a,U_j)} \times \left(r_{u_j,m_i} - \overline{r_{u_j}}\right)}{\sum_{u_j \in N(U_a)} R_{Sim(U_a,U_j)}} \qquad (4)$$

Where P_{u_a,m_i} is the predicted rating of user u_a for movie m_i, $\overline{r_{u_a}}$ is the average rating of user u_a, $N(U_a)$ is the neighborhood set of user u_a computed by equation (3), $R_{Sim(U_a,U_j)}$ is the review similarity between active user u_a and user u_j computed by equation (2).

If the predicted rating of the movie is more than the recommending threshold, the movie is recommended to the active user. All the recommended movies are arranged in the descending order of their predicted rating, so that active user can swiftly get the desired movie to him/her.

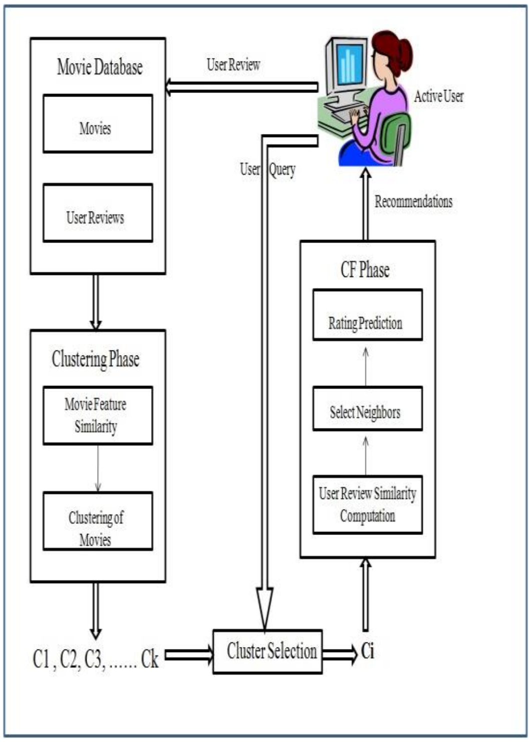


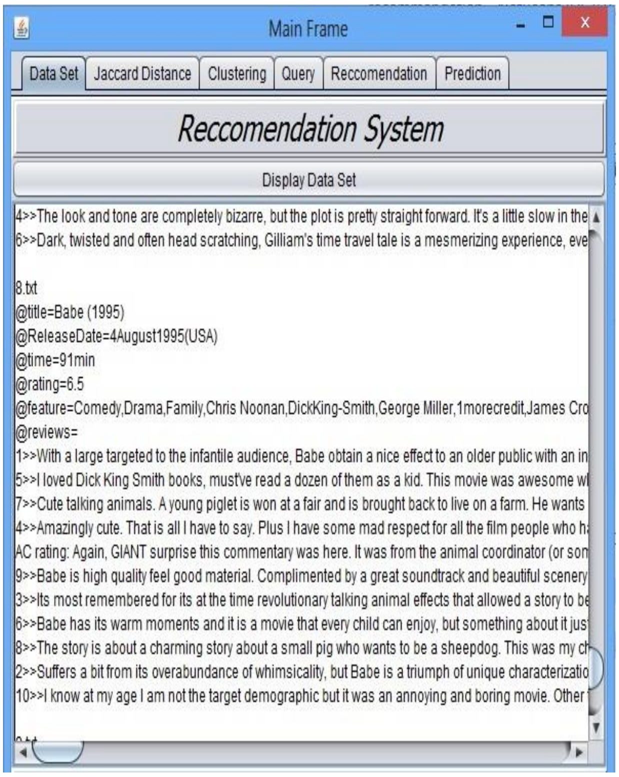
Fig.2. General Architecture Diagram of Proposed System.

Fig. 2 depicts the general architecture of the proposed system. The movies and their reviews are stored in the database. The Clustering is performed on the database. Then the query regarding interested movie is taken by the active user. From the user given query, the appropriate and relevant cluster to the user is selected for further computation. The similarity computation between the active user reviews and the other user's reviews are carried out. Based on the similarity the Top-K users are selected as a neighbor for the active user. The ratings of the candidate movies are predicted and all candidate movies are ranked according to their predicted ratings and presented as recommendations to the active user. Beside this active user can give reviews to the previously viewed movies also. These reviews are stored in the database.

V. Experimental Evaluation

We implement the proposed system and conduct the experiments to analyze and evaluate the accuracy of the proposed system. The evaluation dataset is first introduced in Section 5 (A). The evaluation metric is then introduced in Section 5(B) and finally the comparative accuracy of two different approaches i.e. A rating based versus Review based Recommendations is presented in Section 5(C).

A. Dataset



```
4>>The look and tone are completely bizarre, but the plot is pretty straight forward. It's a little slow in the
6>>Dark, twisted and often head scratching, Gilliam's time travel tale is a mesmerizing experience, eve

8.txt
@title=Babe (1995)
@ReleaseDate=4August1995(USA)
@time=91min
@rating=6.5
@feature=Comedy,Drama,Family,Chris Noonan,DickKing-Smith,George Miller,1morecredit,James Cro
@reviews=
1>>With a large targeted to the infantile audience, Babe obtain a nice effect to an older public with an in
5>>I loved Dick King Smith books, must've read a dozen of them as a kid. This movie was awesome wl
7>>Cute talking animals. A young piglet is won at a fair and is brought back to live on a farm. He wants
4>>Amazingly cute. That is all I have to say. Plus I have some mad respect for all the film people who h
AC rating: Again, GIANT surprise this commentary was here. It was from the animal coordinator (or son
9>>Babe is high quality feel good material. Complimented by a great soundtrack and beautiful scenery
3>>Its most remembered for its at the time revolutionary talking animal effects that allowed a story to be
6>>Babe has its warm moments and it is a movie that every child can enjoy, but something about it jus
8>>The story is about a charming story about a small pig who wants to be a sheepdog. This was my ch
2>>Suffers a bit from its overabundance of whimsicality, but Babe is a triumph of unique characterizatio
10>>I know at my age I am not the target demographic but it was an annoying and boring movie. Other
```

Fig.3. Sample Dataset.

For the experiment we required the data having the movies with its features such as Genre, Actor, Actress etc, and movies having reviews as well as ratings given by different users. The dataset required for our experiment was collected from the RottenTomatoes website. RottenTomatoes is a movie review website which allows users to express their opinions about the movies with a scalar rating as well as text review. The dataset collected includes the Movie Id, Movie Features, User Id along with their text review and scalar rating.

The Data sample took for the experiment has the 25 movies with each movie having minimum 10 different user's reviews and rating associated with it. Thus, our dataset has 250 users reviews and ratings. Our Sample Dataset was collected using the following steps: First we randomly select 25 movies from the mentioned website. Then we select features of movies required for clustering phase and finally we collect the 10 different user reviews along with a scalar rating for each movie.

Fig. 3 shows the sample of the dataset used for the experimental evaluation.

B. *Evaluation Metric*

With respect to find out the effectiveness of the proposed system, we focused on the accuracy of the proposed system. We used Mean Absolute Error (MAE) metric to compare and evaluate the accuracy of two different RSs. MAE is a metric used to calculate how predictions are close to the eventual outcomes. MAE is widely used metric to evaluate the prediction accuracy of RS and it is defined as the absolute difference between the predicted ratings and the actual ratings. MAE is computed using following equation (5):

$$MAE = \frac{\sum_{i=1}^{N} |p_i - q_i|}{N} \tag{5}$$

Where $\{p_1, p_2, p_3, \ldots\ldots p_n\}$ is the set of rating predictions made by proposed system, and $\{q_1, q_2, q_3, \ldots\ldots q_n\}$ is the real rating given by users.

C. *Results and Discussion*

In this section, the results of clustering and recommendation phases are discussed. In the Subsection 1, we show the results obtained in clustering phase. Subsequently, in Subsection 2, we compare the accuracy of the proposed system with existing systems.

1) *Clustering result*

We utilized the AHC clustering algorithm to cluster the available movie dataset into different subclasses. In this phase, first we calculate the Feature similarity between different movies and the obtained feature similarity distance matrix is provided to the AHC as an input. AHC algorithm clusters all movies into different small clusters depending upon the provided distance matrix. AHC algorithm is the hierarchical clustering algorithm which groups the data in tree hierarchy, such that we can adjust the height of the tree to get the variable size of clusters. We can set the height of the hierarchy, so that the obtained cluster size is large enough to hold the sufficient movies. The results of clustering are shown in terms of Dendrogram in Fig. 4. In this the vertical red line shows the height of the hierarchy which we can adjust to set the cluster size as per user need.

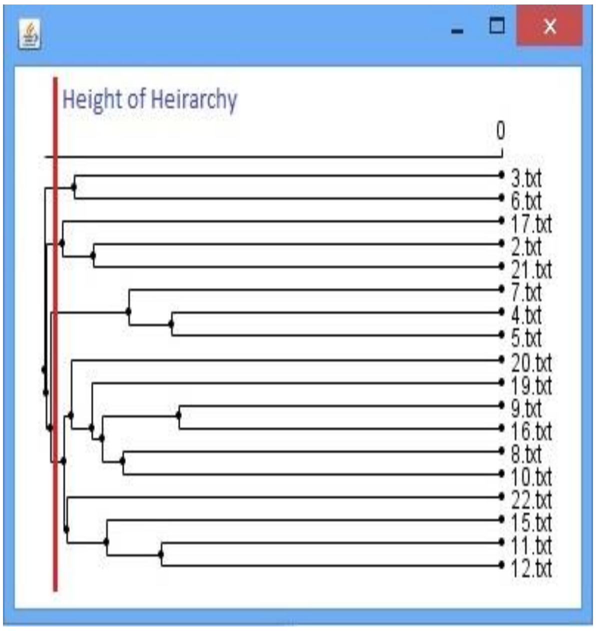


Fig.4. Dendrogram of Clustering.

2) *Predictive accuracy evaluation*

In this section we compared the MAE values obtained for two different approaches used for CF based Recommendation. We compare the MAE values obtained for Rating based RSs to the MAE values obtained for Review based RSs. We computed the MAE values by changing the Top-K recommendation list for two different rating threshold i.e. 2 and 3. We change the values of K. We compute the values of MAE for TOP-3, TOP-5, TOP-8 and TOP-10 movie recommendation lists for each individual approach. The resulted graph of obtained MAE values versus number of Recommendation for both Rating based RSs and Review based RSs is shown in Fig. 5 (a) and Fig. 5 (b).

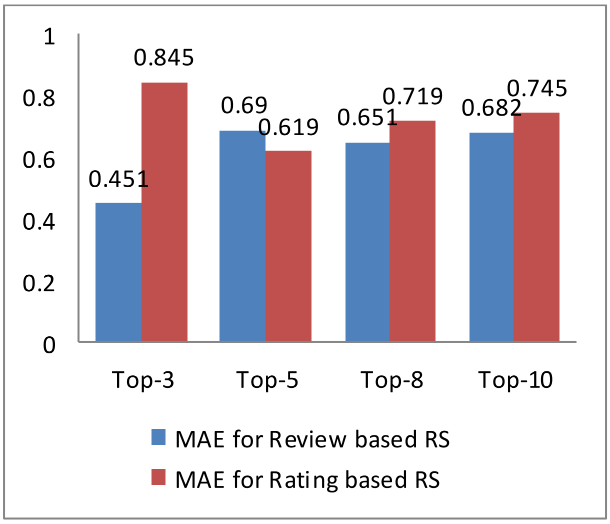


Fig.5(a). MAE Comparison of Review based RS vs. Rating based RS at Rating threshold 3.

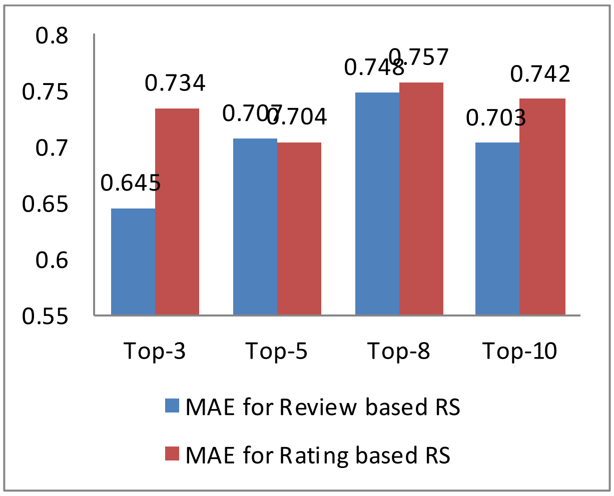


Fig.5(b). MAE Comparison of Review based RS vs. Rating based RS at Rating threshold 2.

Both the figure 5 (a) and 5 (b) shows that, the MAE values of the proposed review based RSs is lower than the Rating based RSs. This shows that, the proposed Review based RSs has more accuracy of recommendation than the traditional Rating based RSs.

VI. CONCLUSIONS

There are various kinds of Recommendation System presents, among which the Collaborative Filtering based RSs are widely used and most successful one. But this CF mainly relies on the rating history of users and uses this rating history to compute the likeminded users. Thus, this CF based on rating history did not consider the reason behind particular rating given by users. As different users may have same rating history with different reasons behind rating, thus likeminded users computed on the basis of pure rating history may inaccurate. So RSs must have to consider these reasons behind rating while computing the likeminded users. In response to this, we have proposed and implemented the RS which consider the user reviews while computing likeminded user set. The proposed system is based on both movie clustering based on movie features and the likeminded user finding on the basis of user reviews. Our proposed method aims at giving the recommendation list of movies with more accuracy and scalability. At the end, the experimental result shows that the proposed method improves the accuracy significantly.

VII. ACKNOWLEDGMENT

I am especially grateful to my guide Prof. Mrs. Manisha P. Mali who has provided guidance and encouragement. I would like to put forward my heartfelt acknowledgement to all my classmates, friends and all those who have directly or indirectly provided their overwhelming support during my project work and the development of this paper.

REFERENCES

[1] Saudagar L. Jadhav, Prof. Mrs. M. P. Mali, "A Survey on Various Approaches Used for Collaborative Filtering Based Recommendation," *International Journal on Advanced Computer Theory and Engineering (IJACTE)*, Vol. 4, Issue-1, Pages 39-44, 2015.

[2] Atisha Sachan and Vineet Richariya, "A Survey on Recommender Systems based on Collaborative Filtering Technique," *International Journal of Innovations in Engineering and technology (IJIET)*, vol. 2 no. 2, pp. 8-14, April 2013.

[3] Reena Pagare and Shalmali A. Patil, "Study of Collaborative Filtering Recommendation Algorithm - Scalability Issue," *International Journal of Computer Applications*, vol. 67 - no. 25, pp. 0975 8887, April 2013.

[4] Robin Burke, "Hybrid Recommender Systems: Survey and Experiments," *User Modeling and User-Adapted Interaction*, vol. 12 no.4, pp. 331-370, November 2002.

[5] J. Bobadilla, F. Ortega, A. Hernando and A. Gutierrez, "Recommender Systems Survey," *Knowledge-Based Systems*, vol. 46, pp. 109-132, July 2013.

[6] G. Adomavicius and A. Tuzhilin, "Toward The Next Generation of Recommender Systems: A Survey of The State-of-the-art and Possible Extensions," *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING*, vol.17 - no. 6, pp. 734749, June 2005.

[7] Shan XU and Junzo WATADA, "A Method for Hybrid Personalized Recommender based on Clustering of Fuzzy User Profiles," *2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, vol. -no., pp. 2171-2177, 6-11 July 2014.

[8] Mukta Kohar and Chhavi Rana, "Survey Paper on Recommendation System," *International Journal of Computer Science and Information Technologies (IJCSIT)*, vol. 3 no. 2, pp. 3460-3462, 2012.

[9] J. Wen and W. Zhou, "An Improved Item-based Collaborative Filtering Algorithm Based on Clustering Method," *Journal of Computational Information Systems*, vol. 8-no. 20, pp. 571-578, 2012.

[10] A. Kohrs and B. Merialdo, "Clustering for Collaborative Filtering Applications," *In Proceedings of Computational Intelligence for Modelling, Control & Automation*, IOS Press, 1999.

[11] S. Saint Jesudoss, "Scalable Collaborative Filtering Recommendations Using Divisive Hierarchical Clustering Approach," *International Journal of Advanced Research in IT and Engineering*, vol. 2 - no. 8, pp. 9-21, August 2013.

[12] S. Gong, "A Collaborative Filtering Recommendation Algorithm Based on User Clustering and Item Clustering," *Journal of Software*, vol. 5 - no. 7, pp. 745-752, 2010.

[13] Xue, Gui-Rong, Chenxi Lin, Qiang Yang, WenSi Xi, Hua-Jun Zeng, Yong Yu, and Zheng Chen, "Scalable Collaborative Filtering Using Clusterbased Smoothing," *In Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 114-121, ACM, 2005.

[14] Ungar, H. Lyle, and Dean P. Foster, "Clustering Methods for Collaborative Filtering," *AAAI Workshop on Recommendation Systems*, vol. 1, pp. 1-16, 1998.

[15] Rong Hu, Wanchun Dou and Jianxun Liu, "ClubCF: A Clustering-Based Collaborative Filtering Approach for Big Data Application," *IEEE Transactions on Emerging Topics in Computing*, vol.2 - no.3, pp.302-313, Sept. 2014.

[16] Hyung and Jun Ahn, "A new similarity measure for collaborative filtering to alleviate the new user cold-

starting problem," *Information Sciences*, vol. 178 - no. 1, pp. 37-51, 2008.

[17] Tsang-Hsiang Cheng, Hung-Chen Chen, Wen-Ben Lin and Yen-Hsien Lee, "Collaborative filtering with user interest evolution," 2011.

[18] Feng Wang and Li Chen, "Recommendation based on mining product reviews preference similarity network," *In Proceedings of 6th workshop on Social Network Mining and Analysis, 2012 ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp.166, 2012.

[19] Asher Levi, Osnat Mokryn, Christophe Diot, and Nina Taft, "Finding a needle in a haystack of reviews: cold start context-based hotel recommender system," *In Proceedings of the sixth ACM conference on Recommender systems (RecSys '12)*, pp.115-122, 2012.

[20] Maria Terzi, Maria-Angela Ferrario and Jon Whittle, "Free Text In User Reviews: Their Role In Recommender Systems," *In Proceedings of the 3rd ACM RecSys10 Workshop on Recommender Systems and the Social Web*, pp. 45-48, October 2011.

[21] S. Meng, W. Dou, X. Zhang and J. Chen, "KASR: A Keyword-Aware Service Recommendation Method on MapReduce for Big Data Applications," *IEEE Transactions on Parallel and Distributed Systems*, vol.25-no.12, pp.3221-3231, Dec. 2014.

Authors' Profiles

Saudagar L. Jadhav is currently pursuing Masters Degree in Computer Engineering from Vishwakarma Institute of Information Technology, Savitribai Phule Pune University, Pune, Maharashtra, India.

Mrs. Manisha P. Mali is working as an Assistant Professor in Department of Computer Engineering, Vishwakarma Institute of Information Technology, Savitribai Phule Pune University, Maharashtra, Pune, India.

How to cite this paper: Saudagar L. Jadhav, Manisha P. Mali, "Pre-Recommendation Clustering and Review Based Approach for Collaborative Filtering Based Movie Recommendation", International Journal of Information Technology and Computer Science (IJITCS), Vol.8, No.7, pp.72-80, 2016. DOI: 10.5815/ijitcs.2016.07.10