Generating Quality Items Recommendation by Fusing Content based and Collaborative filtering

Anand Shanker Tewari, Aleesha S.J.

*Abstract***:** *Recommendation system has become an inevitable part of our life. It has already spread its prominence in various fields like movies, music, news, article recommendations etc. Due to the influence of social media, data is streaming from all over the Internet. Collect the relevant information from chunks of data available has become much difficult. Recommender systems guides in filtering data to get the relevant information. Commonly used recommendation approaches are content based filtering and collaborative filtering. Each approach has its own limitations. The hybrid approach combines the advantages of both the approaches. In this paper, we have tried to enhance the quality of the items recommendation system by fusing both content based and collaborative filtering uniquely. The experimental results are compared with that of other traditional approach using precision and recall evaluation measure. The comparison results show that our approach has 10% better precision for top-10 recommendations than other established recommendation technique.*

*Index Terms***:** *E-commerce, Content based filtering, Collaborative filtering, Hybrid Recommendation System.*

I. INTRODUCTION

Day by day the Internet is getting overloaded with a large amount of data. This makes difficult for users to find out what they exactly need. The relevant ones get meld with that of irrelevant. Recommendation system helps to refine data. Nowadays a lot of researches are going on in the field of recommendation system on how to refine data. There are many approaches to recommend items. Content based and collaborative filtering are the most popular approaches used. *Content based filtering* technique recommends items that are similar in content to the users' and items' profile construction. Construction of users' profile is done with the help of the content of items that were previously rated [1]. For example to suggest the user with items using this approach, first fetches information from the user's profile. According to the profile's information recommendation of items are generated. Constructing a profile to a new user is difficult. Recommending items to a completely new user is thus a difficult task [2].

Collaborative filtering is another popular approach where the recommendations are generated based on the ratings. There are two approaches for collaborative filtering. Like-minded user tends to prefer the same items. The users of the same tastes are grouped together based on the rated items. Items that have vacant ratings are recommended from the

profile of similar users' to the target user. Another approach of collaborative filtering is based on similar items [3]. It recommends items that are similar to the items, the user have rated previously [4].

Hybrid approach combines both the content based filtering and collaborative filtering approach. Combining the advantages of both the approaches makes the recommendation better [5].

The proposed approach is based on the hybrid approach. It has combined the content and collaborative filtering approach. The proposed approach has specialized weights calculation blocks that helps in placing relevant items at top-*n* positions. Thus this approach gives better precision and recall. The following sections are structured as: Section 2 addresses the related works followed by the section 3 briefly explains the proposed approach. Section 4 states practical implementation and evaluation and finally section 5 concludes the paper.

II. RELATED WORKS

The collaborative filtering and content based filtering are few prominent techniques. They are widely used in many commercial applications [6]. In [7] paper authors have used collaborative filtering on items' categories and finds items interestingness, later they combined both these factors to generate recommendations. The Qin, Cao, and Peng have given idea of collaborative filtering recommendations based on weighted items' categories. The approach has tried to handle sparsity problem to improve the recommendations [8]. Mu, Xiao, Tang, Luo, and Yina have introduced new similarity measure, which strongly depends on the distance function [9]. The Gandhi's paper has combined the collaborative filtering with the association rule mining to generate recommendations to improve the quality of the recommendation system [10]. Similarly Tewari and Priyanka have also developed a recommendation system for recommending books using user based collaborative filtering and association rule mining [11]. Kardan and Ebrahimi have proposed a novel approach in hybrid recommendation systems [12]. They have applied association rules mining to trace the best similar users. The Puntheeranurak and Chaiwitooanukool have used items attributes similarities and later use these similarities to adjust the values of the predicted ratings [13]. Statistical Relational Learning can be used for a hybrid approach by using the probabilistic dependencies among the attributes [14]. A news utility model was developed to address the problem of considering clicks as the most effective indicator of real user interests by most of all the existing recommendation system

494

[15]. Social media helps to learn users' interest faster. This technique is used to recommend occupation to the students using their profile, activity and hobbies [16]. A hybrid approach using data mining techniques for tourism recommendation was proposed by Lucas and Joel [17]. They used associative classification which combines classification and association. Movie recommendation by Weia and Shouxian uses tags and ratings [18] for building a social movie network and a preference topic model. The tags are then extracted, normalized to catch up the user preferences.

III. PROPOSED APPROACH

The proposed approach consists of mainly six blocks, each block possessing a unique functionality. The first block called User Profile Construction Block, the second one is Similar User Profile Block, third is Item Rating Prediction Block fourth is Preference Weight Calculation. Fifth is the Common Weight Block. The last block is called Final Prediction block. The detailed description of each block is as follows:

1. User Profile Construction(UPC) Block

The UPC block constructs the user's profile of each user based on the ratings of all the items rated by them. The block also fetches all the features of items, rated by the target user and uses these features to constructs the *user vector*. This *vector* is filled with ratio of *count* of particular feature to the sum of frequencies of all the features used by that user. By this way, the block creates the profile for each user to trace out the user's personal preferences. An example for the *users' vector* table is shown in table 1.

Table 1: Users profile vector table.

Users	Comedy	Romance	.	.	Action
U_1	$\frac{3}{7}$	$\frac{2}{7}$	0	0	$\frac{1}{7}$
U_2	$\frac{3}{9}$	$\frac{3}{9}$	0	0	$\frac{3}{9}$
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
U_m	$\frac{1}{4}$	0	$\frac{2}{4}$	$\frac{1}{4}$	0

2. Similar User Profile (SUP) Block

This block finds the users that have same taste with that of target user. The similar users' profiles are found out using the Pearson correlation similarity ($sim_{u,a}$) as shown in (1).

$$sim_{u,a} = \frac{\sum_{i=1}^{n}(r_{a,i}-\overline{r_a})(r_{u,i}-\overline{r_u})}{\sqrt{\sum_{i=1}^{n}(r_{a,i}-\overline{r_a})^2\left((r_{u,i}-\overline{r_u})\right)^2}} \tag{1}$$

Where $sim_{u,a}$ is the similarity between two users i.e. user u and user a, based on ratings given by both users on all common n items. The similarities between users that maintain a value greater than or equal to certain threshold, sim_t are considered as similar users to the target user. The *SUP* block then places items which all are not rated by the target user but used by his similar users in an *interested_items* list for the target user.

3. Item Rating Prediction (IRP) Block

This block calculates the rating of all items available in the *interested_items* list of the target user. This is to assign priority to the items that have higher rating and are from more similar users' to target user. The prediction calculation is done using (2).

$$R_{u,i} = \overline{r_u} + \frac{\sum_{a=1}^{n}(r_{a,i}-\overline{r_a})\times sim_{u,a}}{\sum_{a=1}^{n}sim_{u,a}} \tag{2}$$

Where $R_{u,i}$ defines predicted rating of item i for the user u. $sim_{u,a}$ is calculated from the *SUP* block. The $\overline{r_u}$ is the mean rating of the target user u and the number users similar to the target user is n.

Table 2: Movie profiles vector table.

Movies	Comedy	Romance	.	.	Action
M_1	1	1	0	0	1
M_2	0	0	1	1	0
.	.	.	.	.	.
.	.	.	.	.	.
M_n	1	0	0	1	1

Once the ratings are calculated, the IRP block then converts all generated items rating in to two values i.e. 0 and 1 [19]. It performs this operation using below steps:

1.) The block arranges all items ratings of each user in the ascending order.
2.) The block then finds all the ratings given by target user which are in the third quartile and above. The starting point of the third quartile defines the *rating_ separat -or_ value*.
3.) The number of predicted items' rating for the target user that are greater than equal to *rating_separator_value* is updated with 1 and rests of the predicted items' rating are set as 0.

4. Preference Weight Calculation (PWC) Block

This block computes the weight of all items available in the target users' *interested_items* list. The items that have more features as in the target user profile, gets higher preference. Example: Suppose, a movie *i* that is in the *interested_items* list of the target user *u*, have genres like *action, thriller, adventurous* and *romance* but the target user have genres like *action, thriller* and *adventurous* in their profile. So, there are three genres in common. The percentage of each genre from the target *user profile vector*, that has entry 1 in *movie profile vector* as shown in table 2, is then added together to find the weight ($w_{u,i}$) of that item as given in (3).

$$w_{u,i} = \sum_{j=1}^{r} f_{u,j} \times b_{i,j} \tag{3}$$

Where $f_{u,j}$ is the jth feature value from the user's vector *u* that matches with the item's vector *i* and $b_{i,j}$ is the binary value of the jth feature of an item's vector *i*. The *r* is the number of common features available in both user's and item's profile vectors.

5. Common Weight (CW) Block

This block finds the preferences of different items among all other users. This helps the proposed recommendation system to place better items for the target user. The CW block helps in rearranging relevant items in the *interested_items* list of the target user. The common weight (CW_i) of an item *i* is calculated by using (4).

$$CW_i = \frac{avg_i}{R_{max}} \times \frac{C_i}{C_{max}} \tag{4}$$

Where avg_i is the items *i* average rating, R_{max} is the maximum rating for the rating scale given. C_i is the total users who gave rating to item i and C_{max} is the maximum C_i value among all items [20].

6. Final Prediction Block

This block takes the inputs from the PWC block, IRP block and CW block. The outputs from all of these blocks together produce a final recommendation for the target user as shown in (5).

$$FRecV_{u,i} = (w_{u,i} \times IRP) + CW_i \tag{5}$$

Where $FRecV_{u,i}$ is the final recommendation value of an item *i* for user *u* and $w_{u,i}$ is the preference weight of user *u* for an item *i* calculated by the *PWC* block. The *IRP* is the output from the *item rating prediction* block and CW_i is the common weight of an item *i*. The items' $FRecV_{u,i}$ values are arranged in decreasing order and are recommended to the target user. Block diagram of the proposed approach is shown in Fig. 1.
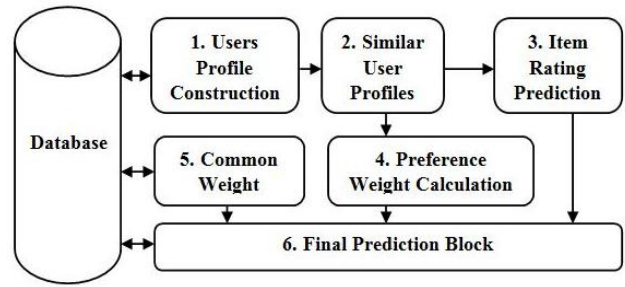

Fig. 1. Block diagram of the proposed approach.

IV. PRACTICAL IMPLEMENTATION AND EVALUATION

We have conducted several experiments to compare the recommendation quality of proposed approach using live dataset.

A. Experimental Setup

The implementation of the proposed system was done in Java 8. Intel Core i7 machine with 7.6 GB RAM is used for experiments. We have used live dataset of http://www.myopinions.in. It has Indian as well as Hollywood movies. The dataset is having 520 realistic users, 1038 movies and 8227 ratings of those movies. Movies rating scale ranges from1 to 5.

B. Experimental Evaluation Methodologies

We compared our proposed approach with *Precision* and *Recall* evaluation measures. These measures are used extensively in the recommendation systems. The qualities of the results are evaluated by these ratios. The ratio between the total relevant items to the total items recommended is the Precision as shown in equation (6) whereas the ratio of the total relevant items that was recommended to the total relevant items in the items library is given by Recall as shown in (7).
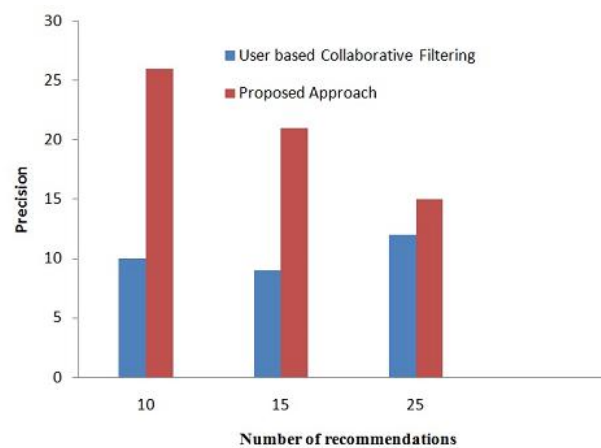

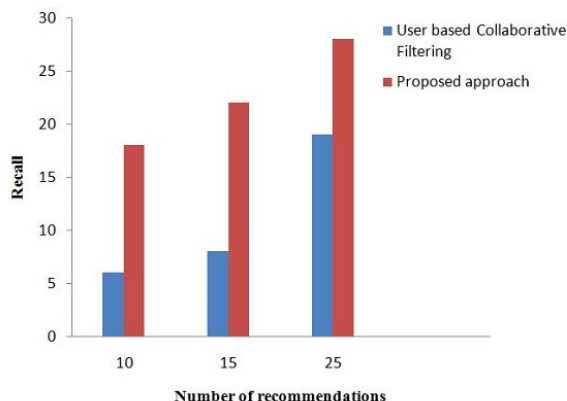Fig. 2. Precision comparison of User based collaborative filtering with the proposed approach.

Fig. 3. Recall comparison of User based collaborative filtering with the proposed approach.

$$Precision = \frac{Number\ of\ Relevant\ Recommendations}{Total\ number\ of\ Recommendations} \quad (6)$$

$$Recall = \frac{Number\ of\ Relevant\ Recommendations}{Actually\ Correct\ Recommendations} \quad (7)$$

C. Practical Evaluation

The experiment is started by implementing the UPC block and then *Similar User* Profile block. The SUP block finds similar users using (1). We calculated the users' similarity for different threshold i.e. sim_t values are as 0.5, 0.6, 0.7, 0.8, and 0.9. The sim_t value at 0.7 has given better precision and recall than rest of the other values. So we have chosen sim_t value as 0.7 and created the interested_items list for users.

To predict the rating of all the movies, in interested_items lists are then fed to Item Rating Prediction block for calculation. In the next step the IRP Block finds the rating_seperator_value of each user. The predicted ratings of movies are then updated with 1 and 0s.

Common Weight block calculates the popularity of all the movies among users using (4). Preference Weight Calculation block finds the weights of different items according to the users' and movies' profile. If there is any match of genre in the movie and user profile then weight is calculated using (3).

The Final Prediction block then finally generates FRecV values for every item in the interested_items list of the target user. These items are recommended in descending order of their FRecV values. The precision and recall value of the generated recommendations are then compared with the established user based collaborative filtering technique. The proposed approach performs better every time than the collaborative filtering approach as visible in fig. 2 and 3. The approach has shown a significant precision improvement of 10% for *top-10* recommendations and 3% for *top-25* recommendations.

V. CONCLUSION

The proposed approach generates better recommendations than traditional recommendation techniques using its unique building blocks. The weight that is introduced in Preference Weight Calculation block enhances to popes the relevant items which were at the bottom of the RS to be in top *n* number. The approach has tried to improve the precision for smaller recommendation list. We compared our approach with that of user based collaborative filtering. The results proves that the proposed approach outperforms than that of user based filtering and help users to fetch the relevant items without much searching throughout the list. This approach requires only users rating and items descriptions for recommendation generation. Thus the proposed RS can be implemented by any movie recommendation system. The work can be improvised by including more data such as tags, demographic information, etc.

REFERENCES

1. Boutemedjet, Sabri, and DjemelZiou. "A graphical model for context-aware visual content recommendation."*IEEE Transactions on Multimedia* 10.1 (2007): 52-62.
2. Pazzani, Michael J. "A framework for collaborative, content-based and demographic filtering." *Artificial intelligence review*13.5-6 (1999): 393-408.
3. Pazzani, Michael J., and Daniel Billsus. "Content-based recommendation systems." *The adaptive web*. Springer, Berlin, Heidelberg, 2007. 325-341.
4. Herlocker, Jon, Joseph A. Konstan, and John Riedl. "An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms." Information retrieval 5.4 (2002): 287-310.
5. Salter, James, and Nick Antonopoulos. "Cinema Screen recommender agent: combining collaborative and content-based filtering." IEEE Intelligent Systems 21.1 (2006): 35-41.
6. Adomavicius, Gediminas, and Alexander Tuzhilin. "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions."IEEE Transactions on Knowledge & Data Engineering 6 (2005): 734-749.
7. Wei, Suyun, et al. "Item-based collaborative filtering recommendation algorithm combining item category with interestingness measure." 2012 International Conference on Computer Science and Service System. IEEE, 2012.
8. Qin, Jie, Lei Cao, and Hui Peng. "Collaborative filtering recommendation algorithm based on weighted item category." 2016 Chinese Control and Decision Conference (CCDC). IEEE, 2016..
9. Yi Mu, Nianhao Xiao, Ruichun Tang, Liang Luo, Xiaohan Yina. "An Efficient Similarity Measure for Collaborative Filtering." Procedia computer science 147 (2019): 416-421
10. Gandhi, Sonali, and Monali Gandhi. "Hybrid Recommendation System with Collaborative Filtering and Association Rule Mining Using Big Data." 2018 3rd International Conference for Convergence in Technology (I2CT). IEEE, 2018.
11. Tewari, Anand Shanker, and Kumari Priyanka. "Book recommendation system based on collaborative filtering and association rule mining for college students." 2014 International Conference on Contemporary Computing and Informatics (IC3I). IEEE, 2014.
12. Kardan, Ahmad A., and Mahnaz Ebrahimi. "A novel approach to hybrid recommendation systems based on association rules mining for content recommendation in asynchronous discussion groups." Information Sciences 219 (2013): 93-110
13. Puntheeranurak, Sutheera, and Thanut Chaiwitooanukool. "An Item-based collaborative filtering method using Item-based hybrid similarity." 2011 IEEE 2nd International Conference on Software Engineering and Service Science. IEEE, 2011.
14. Yang, S., Korayem, M., AlJadda, K., Grainger, T., & Natarajan, S. (2017). Combining content-based and collaborative filtering for job recommendation system: A cost-sensitive Statistical Relational Learning approach. Knowledge-Based Systems, 136, 37-45.
15. Zihayat, M., Ayanso, A., Zhao, X., Davoudi, H., & An, A. (2019). A utility-based news recommendation system. Decision Support Systems, 117, 14-27.
16. Ochirbat, A., Shih, T. K., Chootong, C., Sommool, W., Gunarathne, W. K. T. M., Wang, H. H., & Ma, Z. H.

(2018). Hybrid occupation recommendation for adolescents on interest, profile, and behavior. Telematics and Informatics, 35(3), 534-550.

17. Lucas, Joel P., Nuno Luz, MaríA N. Moreno, Ricardo Anacleto, Ana Almeida Figueiredo, and Constantino Martins. "A hybrid recommendation approach for a tourism system." Expert Systems with Applications 40, no. 9 (2013): 3532-3550.

18. Wei, Shouxian, Xiaolin Zheng, Deren Chen, and Chaochao Chen. "A hybrid approach for movie recommendation via tags and ratings." Electronic Commerce Research and Applications18 (2016): 83-94.

19. Tewari, Anand Shanker, and Asim Gopal Barman. "Collaborative recommendation system using dynamic content based filtering, association rule mining and opinion mining." Int. J. Intell. Eng. Syst 10.5 (2017): 57-66.

20. Tewari, Anand Shanker, Jyoti Prakash Singh, and Asim Gopal Barman. "Generating Top-N Items Recommendation Set Using Collaborative, Content Based Filtering and Rating Variance." Procedia computer science 132 (2018): 1678-1684.

AUTHORS PROFILE

Anand Shanker Tewari has done his B. E. and M.Tech in Information Technology. He has done his Ph.D. in Computer Science and Engineering. Currently he is doing research in recommendation system and published many papers.

Aleesha S. J has done B.Tech in Information Technology. She is currently doing M.Tech in Information Technology from NIT Patna.