Personalized Collaborative News Recommendation

Mansi Sood*
Department of Computer Science, Shyama Prasad Mukherji College
University of Delhi
Delhi, India
Email id: mansii.sood@gmail.com

Dr. Harmeet Kaur
Department of Computer Science, Hansraj College
University of Delhi
Delhi, India
Email id: negi.harmeet@gmail.com

Abstract - With the evolution of World Wide Web many conveniences came to our way, but along with these facilities came some challenges like unlimited information resources and a large corpus of online data. Recommendation Systems have emerged as a solution to this information overload problem. They facilitate users by providing suggestions that effectively prune large information spaces so that users are directed toward items that best meet their needs and preferences. One specific domain is News Recommendation, where thousands of news sources are available online making it difficult for users to find an article relevant to their reading interests. This paper presents an algorithm that takes advantage of predefined categorization done by many online news sources, first to identify user interests and form a relevant user profile out of it and later to recommend news articles that might be of interest to the user. The proposed algorithm adopts collaborative filtering approach to selectively shortlist news articles that should be recommended to users. Outcome is a Personalized News Recommendation System that builds user profiles, requests feedback from users on recommended articles and uses feedback received to continuously monitor dynamically changing user reading patterns, keeping track of articles highly appreciated by users. The paper also presents results received on simulating proposed algorithm on a sample database of college students which indicate good satisfaction level for recommended articles.

Keywords- User Profile, Personalization, News Recommendation System, Preference, Feedback, collaborative Filtering

I. Introduction

With the abundance of electronic information on the World Wide Web and ease to access such information online, users increasingly feel the need of automated systems that can help them navigate this huge information and media landscape. Recommendation Systems have emerged as a solution to this information overload by providing users with personalized recommendation of content suited to their needs and preferences [1, 2, and 3]. Personalization is one way of achieving this selective information filtering by capturing user interests. Most Recommendation Systems take into account the user's usage and liking history as a whole, and summarize this history to be the user's profile [4, 5]. Constructing these user profiles is either done manually, i.e. by asking the users to define them, or in a less obtrusive automated manner, by analysis of usage data [6].

Collaborative filtering (CF) [7, 8] is one of the most successful and widely used technologies in personalization and recommendation systems. Traditionally, Collaborative systems locate peer users with a user profile similar to the current user. These similar users' records are then used to predict the preference value of the current user on a yet to be rated or visited item; or to recommend the top N items user may be interested in. Despite their success and popularity, traditional Collaborative Systems suffer from some well-known limitations like cold start problem [9]. Cold start problem occurs when it is not possible to make reliable recommendations due to an initial lack of ratings of a particular community or group of similar users. Alternative can be to adopt a content based approach, which generates recommendations based on user profile obtained from item features user has already rated, or a hybrid approach which combines both content-based and collaborative filtering technique [10, 11].

A typical example of this information overload is the news industry, which seems to be turning fully online and trying to follow the developments in Web publishing. Most of the news publishers have introduced electronic versions of their content, making it easier for users to read and access latest news updates online [12]. In general, people look for instantaneous access to fresh news updates. When surfing on the internet, user looks for interesting news among a large amount of news articles. Typically, news reading websites retrieve news articles relevant to reading preferences of individual users, and adapt their services based on changes in user's reading interests by employing different recommendation approaches. A challenging problem is how to efficiently select specific news articles from a large corpus of newly published articles, where the selected news items should match the reader's reading preferences. This is referred to as personalized news recommendation [13]. The proposed algorithm will focus on mechanism to selectively choose a set of articles from recently published news articles through collaborative filtering. It can then be used to build a personalized recommendation system that can generate recommendations for users, keeping in mind their reading preferences. The proposed approach extends the algorithm developed in our previous work [14].

II. THE PROPOSED APPROACH TO BUILD NEWS RECOMMENDATION SYSTEM

Algorithm proposed in this paper extends the approach developed in our previous work [14]. It requests a user to perform a one-time signup or registration step specifying their preference for news categories as defined by the news source/provider. Preference Scores received are used to determine the number of recommendations that should be presented to user from each news category. Also, based on these preference scores, user is classified to an existing cluster of users (henceforth, referred to as "Cluster") formed by K-Means algorithm [15, 16]. This classification helps to identify which articles in a category should be recommended to the user. An optional feedback form is associated with every recommended article to understand users' liking about the article. This feedback will be accommodated into existing user profiles to keep track of dynamic reading interests of users. Also, the user feedback received will also be used to rate recommended articles in each news category.

Before directing users towards news articles published on the chosen news source/website, proposed News Recommendation System requires execution of above mentioned steps i.e. it should present a signup form in case of new user or generate recommendations based on existing user profile. Also, it should collect and store feedback of recommended articles to improvise user profiles. This can be done by developing a server system that sits between user and chosen news source. User requests and feedback will be received and processed by this server, news articles will be retrieved from the news source and forwarded to users; i.e. a reverse proxy server system should be developed to ensure that users pass through these above steps before getting access to news articles on the news source/website used.

A. Step 0 – Prerequisite

This step forms clusters on existing user database maintained by the system implementing proposed algorithm. K-Means algorithm [15, 16] is used to form user clusters by using Euclidean distance [16, 17] metric to calculate distance between preferences of two users. User database consists of profiles of all registered users capturing their preferences for each news category (Table I).

Initially when no user is registered on the system, user database is empty. Number of clusters '**C**' to be formed is set just once as administration level input for the system at the time of system deployment. C should be chosen as at least equal to or greater than number of news categories for good results.

Till the time number of registered users on the system is less than 3C (again this magic number 3C is chosen to ensure a good quality grouping of users into initial clusters of similar users), this algorithm behaves just as the algorithm in [14] i.e. it simply adds the user to the database recording their preference scores into user profiles, calculates number of articles to be recommended from each category, and finally presents recent most articles from each category as recommendations for the user. Now, when the number of registered users crosses the threshold of 3C, it performs initial clustering on this database using K-Means clustering algorithm. Re- Clustering on this database can be performed after a fixed number of user sessions conclude or after a fixed no of new users register into the system as decided by the system administration.

Steps listed next explain the scenario when a new user sign ups/registers to the system provided, that 3C or more than 3C users are already registered into the system and initial clustering has also been done, i.e. system is ready with C clusters and their respective centroids.

Sample Illustration

C chosen as 3 for sample illustration; First time clustering will be performed when at least 9 users have already registered to our system.

Initial user preference scores pi (where 1<=i<=3) on a scale of 0 to 10 (0 – lowest, 10 - highest), for three categories, namely, Business, Technology and Sports were manually collected from ten students to form an initial user database for our system. K-Means algorithm was applied to this database to form clusters of similar users.

Preference Score of U1 = 8 (for category Business), 3 (for Technology), 1 (for Sports)

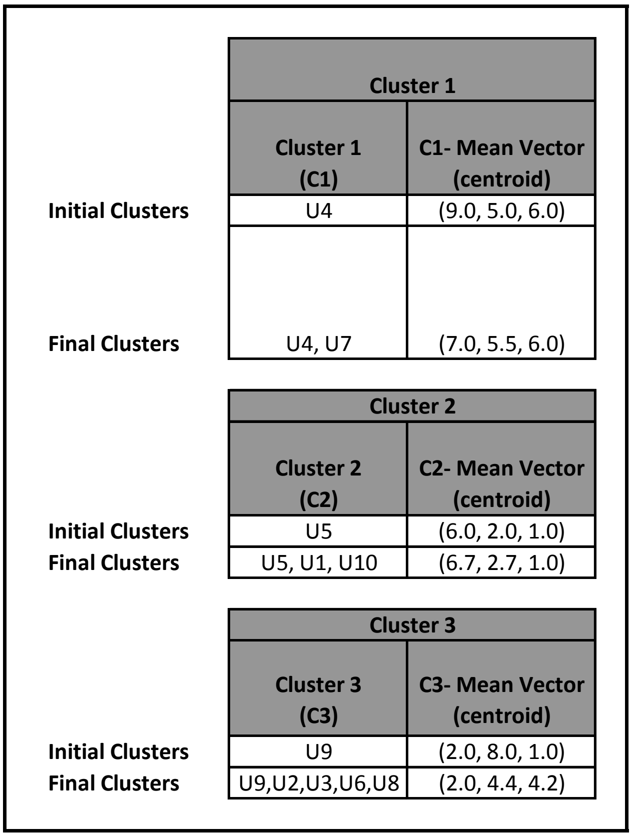
Preference Score of U2 = 2 (for Business), 5 (for Technology), 4 (for Sports)

Euclidean distance between U1 and U2 = $\surd$ $(8\text{-}2)^2 + (3\text{-}5)^2 + (1\text{-}4)^2 = 7$

Initially, U4, U5 and U9 were chosen as 3 cluster centroids. After executing K-Means algorithm, final clusters are shown in Fig. 1

TABLE I. DATABASE OF REGISTERED USERS

Users	**Preference Scores**		
	Business	***Tech***	***Sports***
U1	8	3	1
U2	2	5	4
U3	1	1	5
U4	9	5	6
U5	6	2	1
U6	2	3	5
U7	5	6	6
U8	3	5	6
U9	2	8	1
U10	6	3	1



	Cluster 1	
	Cluster 1 (C1)	**C1- Mean Vector (centroid)**
Initial Clusters	U4	(9.0, 5.0, 6.0)
Final Clusters	U4, U7	(7.0, 5.5, 6.0)

	Cluster 2	
	Cluster 2 (C2)	**C2- Mean Vector (centroid)**
Initial Clusters	U5	(6.0, 2.0, 1.0)
Final Clusters	U5, U1, U10	(6.7, 2.7, 1.0)

	Cluster 3	
	Cluster 3 (C3)	**C3- Mean Vector (centroid)**
Initial Clusters	U9	(2.0, 8.0, 1.0)
Final Clusters	U9,U2,U3,U6,U8	(2.0, 4.4, 4.2)

Figure 1. Initial & Final Clusters

B. Step 1 – Signup

During the one time signup process, new user (say U11) would be asked to subscribe to one or more of the K news categories, (for example Technology, Sports, Business, Entertainment etc.) defined by the news source/website. This would require user to provide Preference Score pi (1<=i<=K) for each category on a scale of 0 to 10, where value 0 indicates lowest preference and 10 indicates highest preference for a given category.

User would also indicate the approximate number of articles to be recommended (N) each time he signs in to the site. N should be greater than or equal to the number of categories subscribed by the user. The actual number of articles recommended by the proposed algorithm will range between N and N+K.

Using these inputs, a personalized profile, capturing user preferences for predefined news categories (as made by used online news source) is created, which will then be used to recommend news articles to the user.

Sample Illustration

User preference scores p_i (where 1<=i<=3), for K = 3 categories, namely, Business, Technology and Sports were collected from the new user (U11). Also, user input was asked to find out approximate number of articles to be recommended.

Available Categories (K = 3): Business, Technology, Sports

User Inputs:

- Preference Scores (pi) of U11 for each category, p1 = 3, p2 = 8, p3 = 6
- Approximate number of articles to be recommended, N = 10

C. Step II – Calculate number of recommendations to be presented from each category

Based on initial preference scores provided by user at Step I, algorithm will calculate the number of articles (n_i) to be recommended from each category, where 1<=i<=K, using the formula stated in equation (1).

$$\text{Ceil}((N*p_i)(p_1+p_2+\ldots+p_K)) \qquad (1)$$

Total recommendations given by this algorithm would be summation of all ni's i.e. $n_1+n_2+n_3+\ldots.+n_k$

Sample Illustration (continued)

$n_1 = \text{Ceil}((N*p_1)/(p_1+p_2+p_3)) = (10*3)/(3+8+6) = 2$

$n_2 = \text{Ceil}((N*p_2)/(p_1+p_2+p_3)) = (10*8)/(3+8+6) = 5$

$n_3 = \text{Ceil}((N*p_3)/(p_1+p_2+p_3)) = (10*6)/(3+8+6) = 4$

Hence, total recommendations given by this algorithm will be 2+5+4 = 11 (ranging between N(10) to N+K (13)).

D. Step III – Classify user to already formed clusters identified at prerequisite step

Based on preference scores received, classify user in one of the clusters, formed at prerequisite step using K-Means algorithm. To classify user into existing clusters Euclidean distance metric [15, 16] can be used and distance between preference scores of user and centroids of all clusters is calculated by (2). Based on this metric, user is mapped to a cluster at minimum distance from preference scores of this user.

Sample Illustration (continued)

Preference score of U11: p_1 = 3 (for Business), p_2 = 8 (for Technology), p_3 = 6 (for Sports)

Euclidean distance between preference scores and cluster C_i with centroid $<c_{i1}\ c_{i2}\ c_{i3}>$ is calculated by

$$\sqrt{}\ (p_1 - c_{i1})^2 + (p_2 - c_{i2})^{\,2} + (p_3 - ci_3)^{\,2} \qquad (2)$$

Based on distance calculated (Table II), U11 is classified to cluster C_3 formed at prerequisite step.

TABLE II. UPDATED P_I BASED ON USER FEEDBACK

New User	Distance to (centroid) of Cluster 1	Distance to (centroid) of Cluster 2	Distance to (centroid) of Cluster 3
U11	4.7	8.2	4.1

E. Step IV – Generating recommendations

Information concluded at previous steps i.e. total number of recommendations to be generated from each category, n_i (Step II) and cluster to which the current user belongs (Step III) will now be used to identify which articles should be recommended to user from each news category so as to achieve a high satisfaction level and matching interests' level for recommended articles.

To perform this step, our algorithm requires a database (shown in Table III) at article level within each category keeping track of highly rated articles in each category. To do this, an article feedback is requested by users every time it is recommended to them. Positive feedback is denoted by +1, negative by -1 and no feedback by zero (as not all users are ready and interested to give feedback for recommended articles). Cluster specific counters are maintained for each article in any category which captures the effective feedback (sum of positive and negative feedbacks, EF) given to the article by users belonging to that particular cluster (Table III). Now to identify articles to be recommended to the current user from a particular news category, article having highest effective feedback in that category is recommended first and then the next highly rated and so on till we reach the n_i i.e. total number of recommendations to be generated from this category.

Sample Illustration (continued)

Following feedbacks were received from users U1 to U10 for various articles under the three categories.

TABLE III. ARTICLE DATABASE MAINTAINED BY ALGORITHM

Category Name	**Article ID**	**EF by C1 users**	**EF by C2 users**	**EF by C3 users**
Business	AB1	7	3	2
Business	AB2	8	4	3
Business	AB3	5	5	4
Business	AB4	6	7	6
Technology	AT1	6	4	8
Technology	AT2	5	4	9
Technology	AT3	7	3	7
Technology	AT4	8	4	5
Technology	AT5	6	4	5
Technology	AT6	4	5	4
Sports	AS1	3	9	6
Sports	AS2	4	7	5
Sports	AS3	5	8	5
Sports	AS4	6	5	4
Sports	AS5	6	6	3
Sports	AS6	5	7	3

Since the user U11 is mapped to cluster C_3 and n_1 i.e. number of recommendations to be generated from Business category is 2, article AB4 and AB3 are recommended to the user, similarly, from Technology and Sports category AT2, AT1, AT3, AT4, AT5, AS1, AS2, AS3, AS4 articles are recommended to the user respectively.

However, if effective feedbacks are not available or some of the articles have same effective feedback values, present recent most articles of that category (to break tie) as recommendations to the user.

F. Step V – Request and Accommodate Feedback

In this step, request the users to give a positive or negative feedback of recommended articles to find out if they were relevant to their taste. Positive feedback carries a value of +1, negative feedback -1 and no feedback carries a value of 0.

Cluster specific effective feedback for each article inside any category is calculated by adding all the feedbacks received for that article by users of that cluster. So add the feedback given by user into corresponding effective feedbacks of recommended articles under the cluster user belongs to.

Also, the feedback given by users is accommodated into user profiles to keep track of their dynamically changing reading interests. To do this, first of all, effective feedback for each category is calculated by adding all the positive or negative feedbacks given by this user for all the articles under that category. Now, recalculate pi i.e. preference score of each news category by adding the original Preference Score pi and calculated effective feedback (as shown in Table IV). Store these new preference scores into user profile so that they can be used next time to generate recommendations for the user (Table IV)

Sample Illustration (continued)

Column 4 in table IV shows the feedback given by user U11. This feedback is used as mentioned above to calculate new preference scores for U11 and to update effective feedback of recommended articles in cluster C_3.

TABLE IV. UPDATED p_i AND EFFECTIVE FEEDBACKS OF RECOMMENDED ARTICLES BASED ON USER FEEDBACK

Category Name	**Article ID**	**Original EF by C_3 users**	**Article Feedback by U11**	**Category Feedback by U11**	**Updated EF of C_3**	**Updated Preference Scores of U11**
Business	AB4	6	1	2	7	5
Business	AB3	4	1		5	
Technology	AT2	9	1	2	10	10
Technology	AT1	8	-1		7	
Technology	AT3	7	0		7	
Technology	AT4	5	1		6	
Technology	AT5	5	1		6	
Sports	AS1	6	-1	1	5	7
Sports	AS2	5	1		6	
Sports	AS3	5	0		5	
Sports	AS4	4	1		5	

Proposed algorithm performs following two additional steps while updating Preference Scores (p_i) for each category:

1. If the updated p_i value for a category increases beyond 10 (say by x), due to positive feedback, subtract x from pi values of all categories.

2. If the updated p_i value for a category decreases below 1 (say by y), due to negative feedback, add y to p_i values of all categories.

It is important to prevent p_i value from becoming 0, to make sure that algorithm generates at least one recommendation for each category where user had indicated an initial interest during signup. Similarly, performance score beyond 10 for a category is taken care of by reducing the scores of other categories to accommodate highly positive feedback for this category. The above two steps, provide robustness to the algorithm, by ensuring that Preference Scores (p_i) for each category remains in range of 1 to 10, even after multiple feedback iterations.

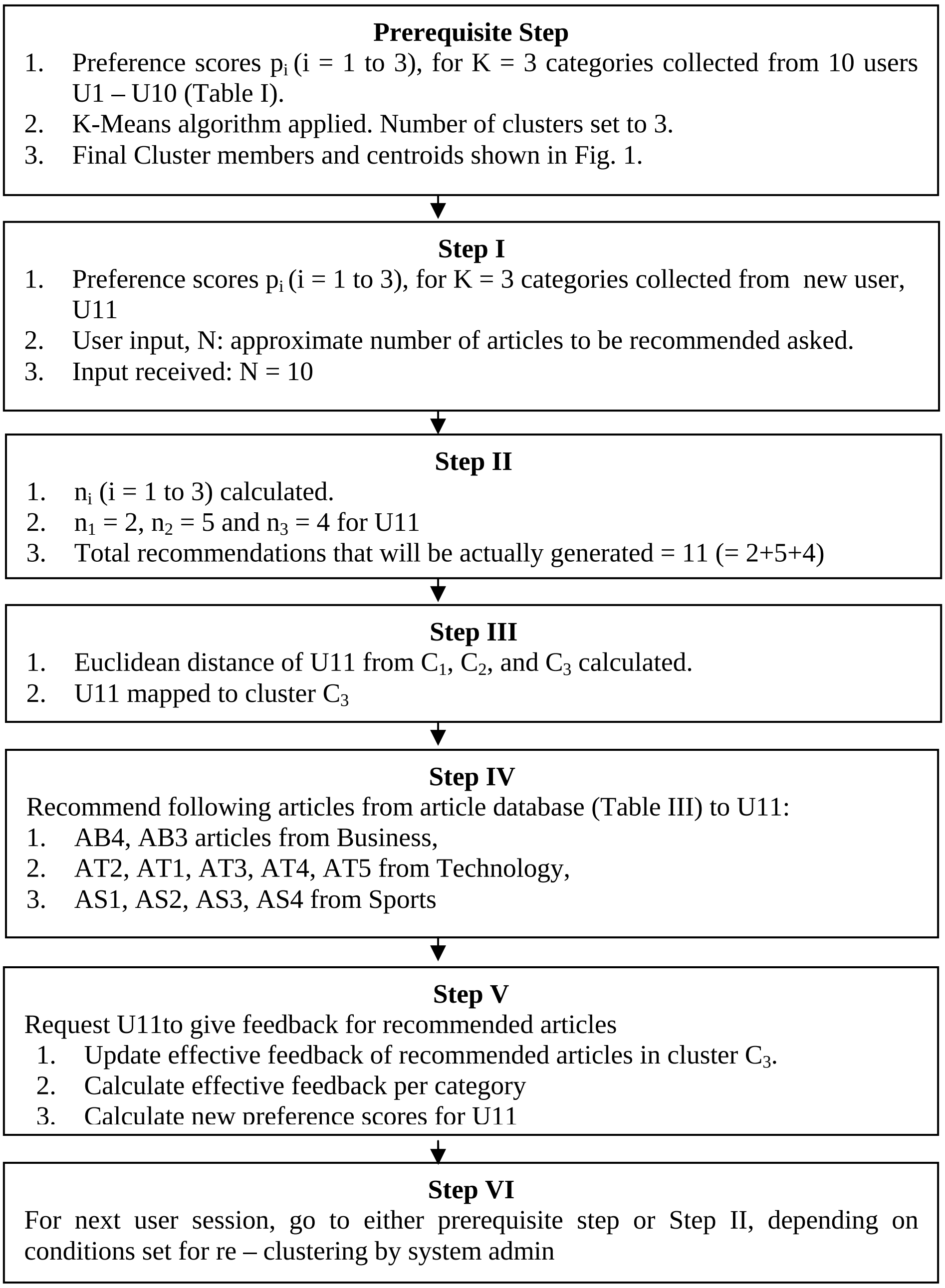
G. Step VI – Future Recommendations

Whenever user logs in for the next time, Step II to Step V are followed again to generate and present news recommendations to the user.

This algorithm has applied collaborative filtering to generate recommendations for the user as on receiving preference scores from the user it tries to match the reading interests of this user with an existing group of users by classifying this user to an existing cluster. This way it finds out other users having similar reading interests and uses their rated and liked articles to generate recommendations for the user.

III. SIMULATION OF PROPOSED ALGORITHM

Proposed algorithm was simulated to estimate its behavior, correctness and robustness on a sample database. Initially ten users registered into the system, followed by the eleventh user for whom all steps of the proposed algorithm have been illustrated in Fig. 2.



Prerequisite Step

1. Preference scores p_i (i = 1 to 3), for K = 3 categories collected from 10 users U1 – U10 (Table I).
2. K-Means algorithm applied. Number of clusters set to 3.
3. Final Cluster members and centroids shown in Fig. 1.

Step I

1. Preference scores p_i (i = 1 to 3), for K = 3 categories collected from new user, U11
2. User input, N: approximate number of articles to be recommended asked.
3. Input received: N = 10

Step II

1. n_i (i = 1 to 3) calculated.
2. $n_1 = 2$, $n_2 = 5$ and $n_3 = 4$ for U11
3. Total recommendations that will be actually generated = 11 (= 2+5+4)

Step III

1. Euclidean distance of U11 from C_1, C_2, and C_3 calculated.
2. U11 mapped to cluster C_3

Step IV

Recommend following articles from article database (Table III) to U11:

1. AB4, AB3 articles from Business,
2. AT2, AT1, AT3, AT4, AT5 from Technology,
3. AS1, AS2, AS3, AS4 from Sports

Step V

Request U11to give feedback for recommended articles

1. Update effective feedback of recommended articles in cluster C_3.
2. Calculate effective feedback per category
3. Calculate new preference scores for U11

Step VI

For next user session, go to either prerequisite step or Step II, depending on conditions set for re – clustering by system admin

Figure 2. Simulation Steps

IV. Simulation Results

The proposed approach extends the algorithm presented in our previous work [14]. Here, we compare the simulation results and behavior of both the algorithms to explain the advantage and improvements of new algorithm over initial one.

Initial algorithm recorded user preferences for various news categories into user profiles, identified number of articles to be recommended from each category based on preference scores received and recommended ‘recent most’ articles from that category. This was reflected in feedback received from users, where feedback was a mix of positive and negative response, sometimes attaining high user satisfaction and at times recording a poor system performance from user’s point of view. However, on receiving a majority of negative responses, system used to change the proportion of number of recommendations to be generated from each category. This was done to accommodate the negative feedback received and improve the quality of recommendations generated. Table V shows the effective feedback received from user U11 during multiple user sessions (4 in count to be specific) on a system implementing this algorithm. Adding up the effective positive and negative feedbacks received for all the three categories during the four user sessions, gives 7 positive and 4 negative feedbacks.

TABLE V. EFFECTIVE USER FEEDBACK RECEIVED

Iterations	**Categories**		
	Business	***Technology***	***Sports***
EF for session1	0	2	1
EF for session2	1	0	-1
EF for session3	-1	0	1
EF for session4	-1	-1	2

The algorithm proposed in this paper also captures user preferences to identify number of recommendations to be generated from each category but while generating recommendations, instead of picking recent most articles, it classifies current user to an existing cluster of other users having similar reading interests as that of the user. It then recommends articles that were highly rated by the chosen cluster of similar users which increases the probability of positive feedbacks from the user. Since recommended articles are the ones, already liked by other users having similar interest patterns, there is a high chance that current user will also like them thereby improving overall user satisfaction and better results in terms of increased positive feedbacks. The results are reflected in table VI showing effective user feedback received from U11, during multiple new reading sessions, for all the three categories on a system implementing this algorithm. Improvement in the results can be observed by looking at the count of positive and negative effective feedbacks received from U11 (16 positive and 4 negative, table VII) where feedback is positive for most of the articles.

TABLE VI. COUNT OF POSITIVE AND NEGATIVE FEEDBACKS FROM U11 ON THE TWO ALGORITHMS

Iterations	**Categories**		
	Business	***Technology***	***Sports***
EF for session1	2	2	1
EF for session2	2	3	1
EF for session3	-1	1	2
EF for session4	1	0	1

TABLE VII. COUNT OF POSITIVE AND NEGATIVE FEEDBACKS FROM U11 ON THE TWO ALGORITHMS

Sum of Feedbacks received in user session	**Initial Algorithm**	**Enhanced Algorithm**
+ve Feedback	7	16
-ve Feedback	4	1

Fig. 3 shows the improvements of algorithm proposed in this paper over the initial algorithm presented in [14] in terms of increased positive and decreased negative feedbacks received in multiple news reading sessions of user U11.

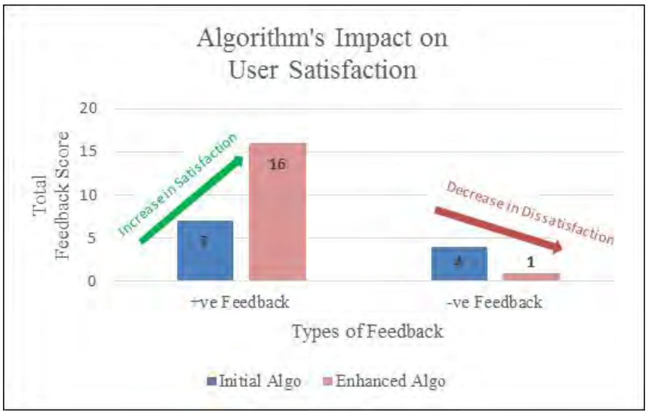


Figure 3. Result graph showing improvement in proposed algorithm over initial

V. CONCLUSION

This paper proposed an algorithm to build Personalized News Recommendation System that captures users' preferences into user profiles, applies collaborative filtering on those profiles to identify users with similar reading interests. It generates news recommendations for users selecting articles that were highly appreciated by a group of users with similar reading interests. It also requests and accommodates user feedback to keep track of dynamically changing user interests thereby identifying articles that are appreciated by them and the ones that they disliked. This algorithm was simulated on a sample database of college students and results were analyzed to identify user satisfaction level by looking at the number of positive and negative responses received. Results indicate that the proposed algorithm achieves good user satisfaction level as the count of positive votes for recommendations were quite higher than the negative votes.

REFERENCES

[1] W. Hill, L. Stead, M. Rosenstein, and G. Furnas, "Recommending and evaluating choices in a virtual community of use", Proceedings of CHI'95.

[2] P. Resnick, N. Iakovou, M. Sushak, P. Bergstrom, and J. Riedl, "GroupLens: an open architecture for collaborative filtering of netnews", Proceedings of the Computer Supported Cooperative Work Conference, 1994.

[3] U. Shardanand, and P. Maes, "Social information filtering: algorithms for automating 'word of mouth'", Proceedings of the Computer-Human Interaction Conference(CHI95), Denver, May 1995.

[4] D. Billsus, and M.J. Pazzani, "A hybrid user model for news story classification", Internatinal Center for Mechanical Sciences, pp. 99–108, 1999.

[5] H.R. Kim, and P.K. Chan, "Learning implicit user interest hierarchy for context in personalization", Proceedings of IUI, pp 01–108, 2003.

[6] P. Bedi, H kaur, B gupta, J. Talreja, and M. Sood, "A Website Recommender System based on Analysis of User's Access Log", Journal of Intelligent Systems, Volume 18, Issue 4, pp 333–352, March 2011.

[7] B. Mobasher, X. Jin, and Yanzan Zhou, "Semantically enhanced collaborative filtering on the web", First European Web Mining Forum, EWMF 2003, Cavtat-Dubrovnik, Croatia, September 22, 2003.

[8] J. Herlocker, J. Konstan, A. Borchers, and J. Riedl, "An algorithmic framework for performing collaborative filtering", Proceedings of the 22nd ACM Conference on Research and Development in Information Retrieval (SIGIR'99), Berkeley, CA, August 1999.

[9] B. M. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Analysis of recommender algorithms for e-commerce", Proceedings of the 2nd ACM E-Commerce Conference EC'00, Minneapolis, MN, October 2000.

[10] R. Burke, "Hybrid recommender systems: survey and experiments", Journal of User Modeling and User-Adapted Interaction, Vol 12, Issue 4, November, 2002.

[11] R. Burke, P. Brusilovsky, A. Kobsa, and W. Nejdl, "Hybrid web recommender systems", The Adaptive Web, LNCS 4321, pp. 377 – 408, 2007.

[12] G. Paliouras, A. Mouzakidis, V. Moustakas, and C. Skourlas, "PNS: A Personalized News Aggregator on the Web", studies in Computational Intelligence (SCI), pp. 175–197, 2008.

[13] K.G. Saranya, and G.Sudha Sadhasivam, "A Personalized Online News Recommendation System", International Journal of Computer Applications, Vol. 7-No.18, November 2012.

[14] M. Sood, and H. Kaur, "Preference Based Personalized News Recommender System", International journal advanced Computer Research, 2014, in press.

[15] R. Sharma, M. A. Alam, and A. Rani, "K-Means Clustering in Spatial Data Mining using Weka Interface", International Conference on Advances in Communication and Computing Technologies, 2012.

[16] C. Zhang, and Z. Fang, "An Improved K-Means Clustering Algorithm", Journal of Information & Computational Science, pp 193-199, 2013.

[17] S. Zhong and J. Ghosh, "Generative model-based document clustering: A comparative study," Knowledge and Information Systems: An International Journal, February, 2005.