# Combining Personalized and Non-Personalized Recommendations

**Apoorva Joshi[#1], Apoorva Patankar[#2],Gurpreet Kaur Chabada[#3], Apurva Sawant[#4]**

#1 C/601 Suncrest, Accolade Society, Green Rd, Louis Wadi, Thane (W) 400604, India, 9819697762.

#2 102 Geet Govind, Samta Nagar, Pokhran Rd No.1, Thane (W) 400606, India, 9819416605,

#3 A-6/504 Happy Valley, Manpada, Thane (W) 400607, India, 9819426663,

#4 A-101, AkshayDeep CHS, Mhada Colony, Mulund (E), Mumbai 400081, India, 9821311081.

## ABSTRACT

Recommender systems play a significant role in E-commerce businesses by attracting more number of customers by providing them information related to their context which helps to narrow down their search. Many filtering techniques are used to anticipate what the user is exactly in search for. Widely used techniques are content based filtering and collaborative filtering. Both of these techniques when used alone have certain limitations since collaborative filtering systems do not explicitly incorporate feature information, content-based systems do not necessarily incorporate the information in preference similarity across individuals. Hence we have used a hybrid technique which is a combination of both techniques and it gives better performance by overcoming limitations of each. Our proposed approach overcomes problems like overspecialization and cold start problem of content based filtering techniques and data sparsity, gray sheep problems of collaborative filtering techniques. We will use an algorithm to rank the recommended items for a user by assigning ratings to each item which will be a weighted combination of ratings coming from content based filtering, user based collaborative filtering and also non personalized recommendations like ratings of highly popular items. Our approach also extracts user preferences and uses the user's browsing activity on the site to understand his changing tastes so that they can be incorporated while making future recommendations.

**Key words:** Collaborative Filtering, Content Based Filtering, Hybrid Model, Recommender Systems, Memory based Filtering, Model Based Filtering

**Corresponding Author:** Apoorva Joshi, Apoorva Patankar, Gurpreet Kaur Chabada, Apurva Sawant

## INTRODUCTION

With the advance of Internet many E-Commerce businesses have been established over the years as the products sold online reach more prospective consumers. A large number of E-Commerce businesses have adopted the strategy of using recommender systems to increase user's experience while browsing on their site. Recommender systems improve E-Commerce sales by turning browsers into buyers, improving cross selling and increasing customer loyalty [1]. When a user searches for any product on a website he does not immediately get the desired one out of thousands of products present in the website's database. A recommender system aids the user in reducing efforts from his side in his quest of finding the products he is interested in. Recommender systems provide personalized and non- personalized recommendations to an active user. Personalized results are customized according to each user by taking into consideration his demographic features, his behavioral pattern and item attributes. Non-personalized recommender systems recommend products to customers based on what other customers have said about the products on an average. The recommendations are independent of the customer, so each customer gets the same recommendations. A lot of research is going on in increasing the accuracy and efficiency of recommender systems and its success lies in the extent to which users have been satisfied by the results [2]. Furthermore, the recommender industry will be larger, and recommender technologies will be more pervasive than the search industry and search technology in the long term.

Several techniques have been proposed in the literature for recommender systems, and majority of them use the ratings given by the user for some of the items that he views. The ratings can be asked from the user explicitly or can be given implicitly by the system by analyzing the user's browsing activities and history. These ratings are then used to predict the ratings of the remaining unrated items in the user item matrix. So the basic idea which recommender systems employ is to first collect ratings from the user for some of the items and then use those ratings for predicting the ratings that the user would give to the unrated items. The major approaches for recommender systems can be classified into collaborative techniques, content-based techniques and hybrid techniques that are a combination of both ([3], [4], [5], [6]).

Each filtering technique comes with certain limitations and benefits of its own hence by combining both we can compensate limitations of each technique. Content based technique when used in isolation tends to give overspecialized results because of which the chances that a user will get recommendations outside his domain of preferences or browsing history are less. To compensate for this we can infuse some user based collaborative characteristics by finding users similar to the active user and provide recommendations to the active user based on the items viewed or rated by the similar user. We propose a approach where we will collect the ratings coming from content based filtering, user based collaborative filtering and the non personalized techniques where each technique will be assigned a confidence measure that will show the importance of their contribution to the final rating. The value of the parameters assigned to each technique will vary in accordance to the number of ratings we have per user. This technique will address problems like cold start problem and overspecialization problem of content based filtering and also data sparsity problem and grey sheep of the collaborative filtering technique.

In this paper we discuss benefits of using the proposed technique by applying it to dataset used for a website that suggests interior and furnishing items to users. We present some information regarding the current techniques used along with their limitations in Section 2. In Section 3, we discuss the proposed approach in detail. In Section 4 we discussed the outcomes of this technique. We reach to a conclusion in Section 5.

## MATERIALS AND METHODS

### Related Work

#### *A. Content based Filtering*

In such methods ratings are taken as an input from the user implicitly or explicitly and are used to generate a user profile which describes his taste. Demographic features of the user such as job, location, sex are also taken into account. His browsing activity on the site in terms of his navigation patterns, viewing patterns and browsing history is also used to understand his interests. All the unrated items are compared with this profile and the most similar ones are presented to the active user. Content-based recommendation methods can be applied in domains where it is possible to describe item by features. In domains such as movies, videos, or music, features extraction is hard to achieve [7]. The content-based approach to recommendation has its roots in information retrieval and information filtering research [6]. The limitations faced by this technique are as follows:

#### *1) Limited Content Analysis*

The recommender systems that use content based filtering have to explicitly associate feature information with the items that they recommend. Therefore, in order to have a sufficient set of features, the content must either be in a form that can be parsed automatically by a computer (e.g., text) or the features should be assigned to items manually. Another problem with content based filtering is that two items cannot be distinguishable if they are associated with the same set of features [7].

#### *2) Overspecialization*

Once the user profile is established, it may happen that only those items whose content information is similar to the user's interests will be recommended to that user. This means that an item which is not scoring well against the users profile will not be considered for recommendation and the user will keep getting recommendations from within his domain of interest only.

#### *3) New User Problem*

When a new user arrives at the site it, naturally it will take time for the recommender system to collect information about his navigation patterns, browsing activities, his rating and viewing history etc. Thus the recommender system will be able to produce reliable and accurate recommendations only when the user has viewed or rated a sufficient number of items. Therefore, a new user, having very few ratings, would not be able to get accurate recommendations [8].

### *B. Collaborative Filtering*

Given a domain of items, users can give their ratings to items they have tried before. The recommender can then compare the active user's ratings to those of other users, find the most similar users based on some criterion of similarity, and recommend items that similar users have liked in the past. We also sort collaborative filtering methods into two types namely Heuristic based (memory based) method and model-based method [3,4]. Memory based algorithms, which we apply in this paper, utilize the entire user-item database to generate a prediction. These systems employ statistical techniques to find a set of users, known as neighbors, that have a tendency to agree with the tastes target user (i.e. these set of users rate a set of items similarly). Once a neighborhood of users is formed, these systems use different algorithms to combine the preferences of neighbors to produce a prediction or top-N recommendation for the active user. Top-N recommendations can be user based or item based. These methods are notably deployed into commercial systems such as Amazon because they are easy to implement and highly effective. However, there are several limitations of memory based techniques such as the fact that the similarity values are based on common items and therefore are unreliable when data is sparse and therefore common items are few. To achieve better prediction performance and overcome shortcomings of memory based algorithms, model-based approaches have been investigated. Model based techniques use the pure rating data to estimate or learn a model to make predictions. The model can be a data mining or a machine learning algorithm. Well-known model-based techniques include Bayesian belief nets models, clustering models, and latent semantic models [3, 4]. Top N recommendations in memory based techniques can be given by user based or item based collaborative techniques.

#### *1) User based top N recommendation algorithms*

User based top recommendation algorithms firstly identify the most similar users (nearest neighbors) to the active user using certain similarity measures. After most similar users have been discovered, their corresponding rows in the user item matrix are aggregated to identify a set of items purchased by the group together with their frequency. This set is then used by to algorithm to recommend the top most frequent items that the active user has not purchased. There are several algorithms such as Pearson Correlation Similarity, Spearman Correlation Similarity, Tanimoto Coefficient Similarity, Loglikelihood Similarity and Euclidean Distance Similarity that have been used to compute similarities between users. User-based top-recommendation algorithms have limitations related to scalability and real time performance.

#### *2) Item based top N recommendation algorithms*

Item based collaborative filtering uses ratings of items to find item based similarities. In this approach we find the similar items based on the users who rated both. To calculate similarity between items, users who rated both items are isolated and a similarity computation algorithm is applied eg, Euclidean Distance Similarity.

### C. Challenges of Collaborative techniques:

#### 1) Data Sparsity

In any recommender system, the initial number of ratings obtained is very small as compared to the number of ratings that are yet to be predicted. It is essential to produce accurate and reliable ratings for the unrated items using these small set of ratings. One way to overcome the problem of rating sparsity is to use user profile information when calculating user similarity. That is if the ratings of the users are falling short to prove that they both are similar then we can use their user profiles to find similarity between them.

#### 2) Cold Start Problem

This problem is also called collectively as the new user problem and new item problem in many papers. New items are added regularly to recommender systems. Collaborative systems rely solely on users' preferences to make recommendations. Therefore, until the new item is rated by a substantial number of users, the recommender system would not be able to recommend it. Just as the items new users also log into the website and it takes some time to accumulate some information about them. It won't be possible to establish similarity between two users until we have enough information to show that they are similar. This problem is addressed by hybrid techniques discussed below.

#### 3) Gray sheep

Gray sheep refers to the users whose opinions do not consistently agree or disagree with any group of people and thus do not benefit from collaborative filtering.

#### 4) Shilling attacks

Since recommender systems provide an open platform for public to express their opinion, many people take undue advantage and start giving ratings that can be misleading. People may give tons of positive recommendations for their own materials and negative recommendations for their competitors. Such malpractices should be discouraged in order to maintain accuracy of the recommendations [8].

### D. Hybrid Techniques

Several recommendation systems use a hybrid approach by combining collaborative and content-based methods, which helps to avoid certain limitations of content-based and collaborative systems. It makes the recommendation process more robust and efficient. Hybrid techniques can be classified into seven classes namely: weighted where each of the recommendation approaches makes predictions which are then combined into a single prediction [6]; switching where one of the recommendation techniques is selected to make the prediction when certain criteria are met; mixed in which predictions from each of the recommendation techniques are presented to the user[5]; feature combination where a single prediction algorithm is provided with features from different recommendation techniques; cascade where the output from one recommendation technique is refined by another; feature augmentation where the output from one recommendation technique is fed to another, and meta-level in which the entire model produced by one recommendation technique is utilized by another[4].

## Proposed Approach

In this section we present an approach where we provide a combination of personalized recommendation and non personalized recommendation. In the personalized recommendation technique we use hybrid approach where we combine content based and collaborative filtering technique. The reason we infused non personalized recommendation approach is because the popular or trending items in the database carry value and are significant. Any user would want to know about trending items if he wishes to come out of his comfort zone and also trending items carry the power to influence his interests in case the personalized recommendations become too overspecialized. Using hybrid techniques will address the gray sheep problem of collaborative filtering techniques to an extent. In the collaborative filtering technique we utilize user based filtering technique that belongs to memory based method. Each technique is explained below.

The user first registers onto the site and during the registration process some demographic information will be extracted from the user. The user will be shown a set of images where each image will depict a genre of the furnishing item. Each image has a set of tags associated with it which describe the genre depicted by it. The user will click those images which appeal to him the most. The tags will be extracted and stored as his preferences in order to understand his interests. As he browses the site, views and rates the items, his preferences will be continuously updated to record his changing tastes.

### *A. Content based filtering*

This technique finds the set of items previously rated or viewed by the active user and selects the items that are similar to the target item using a similarity measure. E.g.: Each item in the database has tags associated with it which describe the feature information about the item. Also the users profile also contains his preferences, browsing history ratings, clicks etc. We used the KNN (nearest neighbor) algorithm to find the most similar items to target item. To calculate the similarity measure we employed Pearson's Correlation since it deals with the differences in rating scale by subtracting the average user's ratings from each co-rated pairs of items [9]. Also Pearson's Correlation helps in addressing the scalability problem.

$$\hat{r}_{u,i}=\hat{r}_u + \frac{\sum_{v=1}^{N} r_{v,i}-\hat{r}_v \times P_{u,v}}{\sum_{v=1}^{N} P_{u,v}}$$

Where $r_{u,i}$ is the predicted rating of the user u on the item i;
$\hat{r}_u$ is the mean rating given by the user u;
$P_{u,v}$ is the Pearson correlation similarity between users u and v;
N is the number of users in the neighborhood;

### *B. Collaborative Filtering*

We use the neighborhood based approach which is commonly used. In this approach, the users having the same interests as the active user are found out. Then the rating that the active user *u* would give to item *i* is calculated based on the ratings of similar users. Such approach depends strongly on the number of neighbors that rate the item *i* and share similar ratings profile as the user *u*. The neighbors of the user *u* are computed using a similarity measure. Formally, let *u* and *v* be two users, and $RI_{u,v}$ let be the set of all items co-rated by both users *u* and *v*. In the cosine based method, users *u* and *v* are treated as vectors in the $|RI_{u,v}|$−dimensional space [9].Therefore, the similarity is computed as follows:

$$sim(u,v) = cos(\vec{u},\vec{v}) = \frac{\vec{u}\cdot\vec{v}}{\|\vec{u}\|_2 \times \|\vec{v}\|_2} = \frac{\sum_{i \in RI_{uv}} r_{u,i} \cdot r_{v,i}}{\sqrt{\sum_{i \in RI_{uv}} r_{u,i}^2} \cdot \sqrt{\sum_{i \in RI_{uv}} r_{v,i}^2}}$$

### C. Rating prediction

The most important aspect of our approach is combining content based, collaborative and non personalized recommendation approach. Each algorithm independently predicts the rating that user *u* would give to item *i*. We combine linearly the ratings predicted by these algorithms by taking into account the contribution of each algorithm with respect to the final rating prediction. To take into account the difference in the contribution of each predictor in the final rating prediction, we associate a parameter to each predictor. In our approach we emphasize more on ratings predicted by content based algorithm since we already have his preferences stored before he starts browsing. These preferences can be used to find those items that score highly against his profile. If the active user has already rated a number of items similar to the target item *i*, then the content-based algorithm will perform better than the other algorithms. In the same way, if the user *u* has a great number of neighbors that have rated the target item *i*, then the collaborative filtering will perform better than the other algorithms. In those cases where the system doesn't have enough information of the active user's previous ratings then the collaborative filtering won't work well. In such cases using information regarding highly rated items or popular items along with a little knowledge about his preferences we can achieve a satisfactory rating . The final rating prediction is given by the following formula:

$$\hat{r}_{u,i} = \frac{\alpha NP_{u,i} + \beta CF_{u,i} + \gamma CB_{u,i}}{\alpha + \beta + \gamma}$$

To compute the value of each parameter, we should use a function $\Psi(n)$ that gives value 1 for big values of user's ratings $n$ ($n = |RIu|$) and a small value for small values of $n$. The sigmoid function satisfies these constraints for $\Psi(n)$. Therefore, the parameters $\alpha$, $\beta$ and $\gamma$ can be computed using the sigmoid function of $n$ as follows:

$$\gamma = \frac{1}{1 + e^{\frac{-n}{2}}}$$

$$\alpha = \beta = \frac{1}{1 + e^{\frac{-n}{2}}}$$

The parameters $\alpha$, $\beta$ and $\gamma$ represent a sort of confidence given respectively to recommendation algorithms NP (non-personalized), CB (content based) and CF (collaborative filtering). The greater the number of user's ratings, the more confidence will be given to that algorithm. The values of these parameters change dynamically depending on the available ratings expressed by the user $u$.

## Experimental Results

In this section we show the experimental study conducted on our website which recommends interior and furnishing items. We collected the user's demographic information also. Our dataset consisted of 640 items and 250 users and around 6000 initial ratings. A set of 24 images showcasing different combinations of different interior designs were shown to our users. The user then selected the images which appealed to him the most. The users selected on an average 5 images. There were tags associated with each image which described the genres of the interior designs. After extracting those tags from the users, ratings were given by our system implicitly to the items that belonged to the same genres as the designs which the users preferred. The users were also given the option of rating items explicitly while browsing the site if some had appealed to them at that time. We then applied our algorithms to predict the ratings of unrated items. We use the Root Mean Squared Error (RMSE) to measure the error in our recommender system as follows:

$$RMSE=\sqrt{\frac{\sum_{(u,i)\in TestSet}(r_{u,i}-\hat{r}_{u,i})^2}{|TestSet|}}$$

$r_{u,i}$=actual rating
$\hat{r}_{u,i}$=predicted ratings

The smaller the value of *RMSE*, the more precise a rating prediction.

The neighborhood size (*K* parameter) plays an important role in the prediction results. We varied the number of neighbors from 10 to 100 and computed the *RMSE*. We found out that it was stable when value of k was 50 and hence chose 50 as the neighborhood size. We applied content based and collaborative based in isolation and also applied a combination of both excluding the non personalized recommendations. It was found out that users were satisfied much more when a combination of all the three methods where as used as compared to the isolated usage of each.

The content based filtering contributes greatly in the results of our approach in the case of users with a large number of ratings, which increases the value of $\gamma$ and decreases the two other parameters $\alpha$ and $\beta$. In case there are not much ratings available the values of $\alpha$, $\beta$ and $\gamma$ are evened out. Cold start problem was not a big challenge since we had a basic set of preferences of the users and also had their demographic information stored in the database. The values of $\alpha$, $\beta$ and $\gamma$ are not predefined and depend heavily on the amount of ratings made by the users. The dynamic way in computing the parameters $\alpha$, $\beta$ and $\gamma$ makes our approach flexible, adaptable and more practical in real systems. Once the predictions were computed we showed the user a ranked list of top N recommendations.

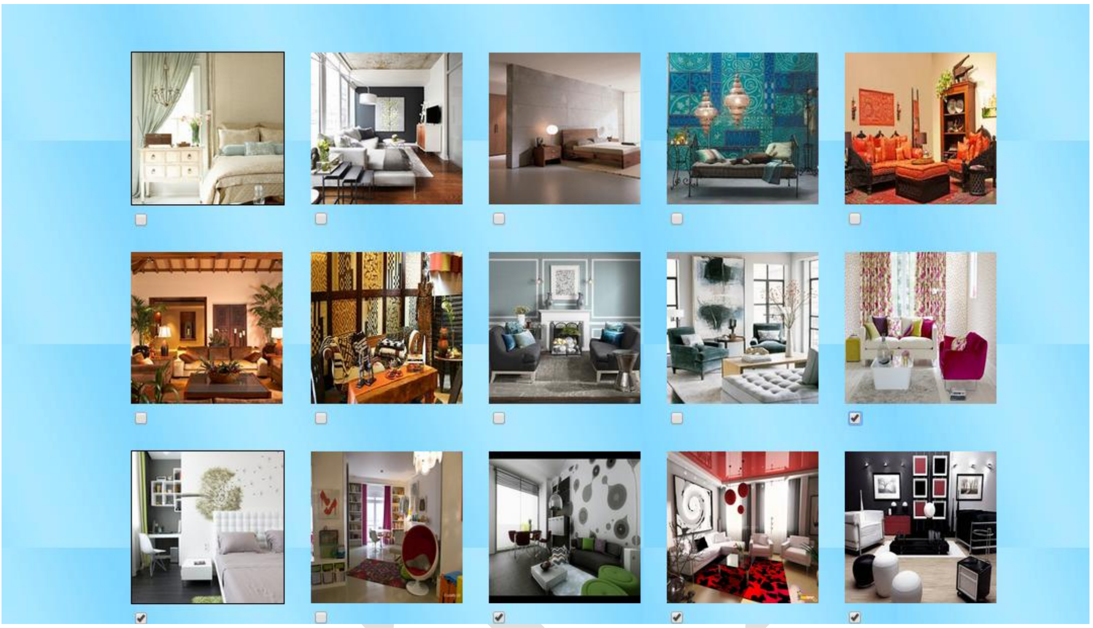

Fig 1: Set of images displayed to user during registration process to understand his interests

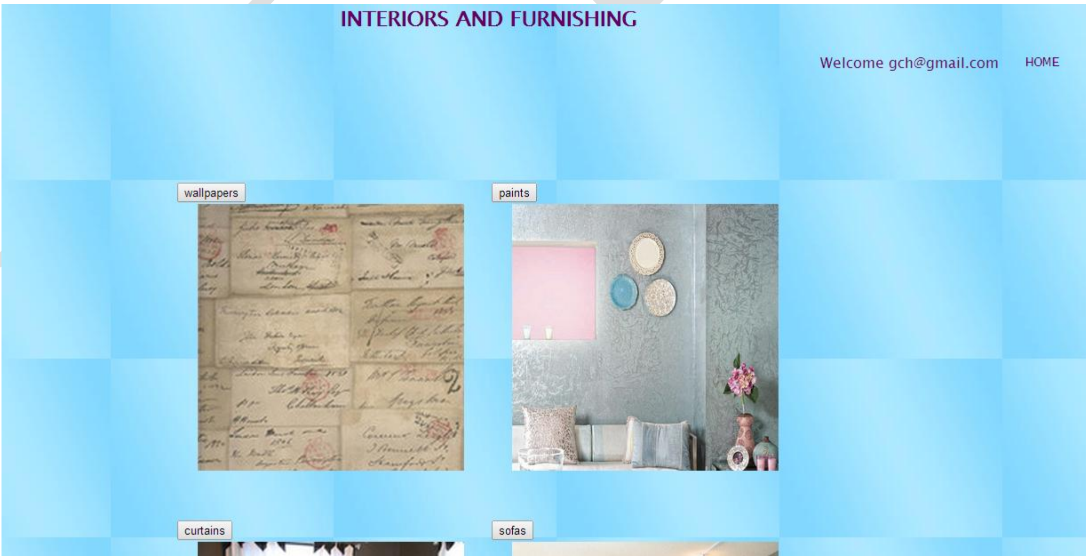


Fig 2: Home Page

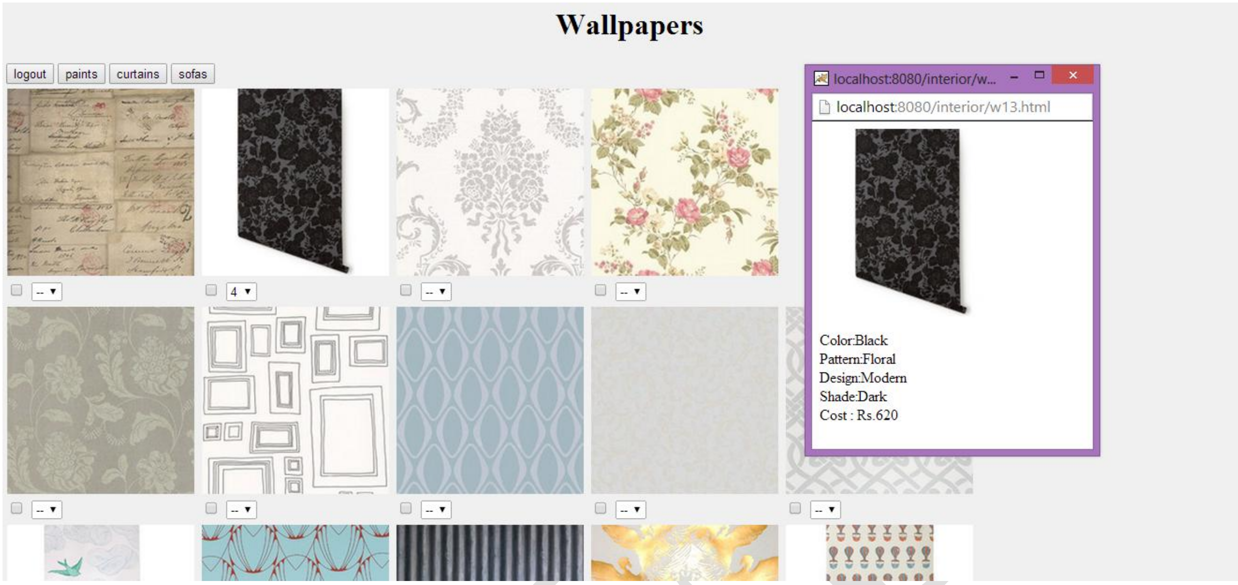


Fig 3: Recommendations shown to user for wallpapers

## CONCLUSION

In this paper we introduced an efficient hybrid approach for recommender systems. We combined personalized and non personalized recommendation techniques to predict ratings. We attributed a confidence measure to each recommendation algorithm. This confidence measure is computed according to the number of ratings available for a particular user. We propose the concept of introducing non personalized results to combination of content based and collaborative filtering techniques since a significant improvement in the performance of recommender systems is in terms of user satisfaction. By introducing non personalized recommendations we protected the system from the effects of overspecialization to an extent. The problem of data sparsity and cold start problem was efficiently solved.

There is still a lot of scope of improving the proposed methods and making it more efficient. We want to make the recommender system more scalable by forming clusters of users based on different categories and similarities and test it on different data sets.

## REFERENCE

[1] Ya-Yueh Shih, Duen-Ren Liu, “Hybrid recommendation approaches: collaborative filtering via valuable content information,” in Proceedings of the 38th Hawai International Conference on System Sciences , 2005.

[2] Manoela Ilic, Jo˜ao Leite, Martin Slota, “Scalable dynamic user preferences for recommender systems through the use of the well-founded semantics,” in IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, 2008

[3] Dilek Tapucu, Seda Kasap, Fatih Tekbacak, " Performance comparison of combined collaborative filtering algorithms for recommender systems," in IEEE 36th International Conference on Computer Software and Applications Workshops, 2012.

[4] Gediminas Adomavicius, Alexander Tuzhilin, "Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions," in IEEE Transactions On Knowledge And Data Engineering, Vol. 17, No. 6, June 2005.

[5] Huan-Yu Lin, Jun-Ming Su, Yi-Li Liu, Jin-Long Li, "OSCAR: an online scalable adaptive recommender for improving the recommendation effectiveness of Entertainment Video Webshop ," unpublished

[6] Gilbert Badaro , Hazem Hajj , Wassim El-Hajj , Lama Nachman, " A hybrid approach with collaborative filtering for recommender systems" unpublished

[7] Vipul Vekariya, G. R. Kulkarni, "Hybrid recommender systems: content-boosted collaborative filtering for improved recommendations," in International Conference on Communication Systems and Network Technologies, 2012

[8] Cong Li and Zhigang Luo " Detection of shilling attacks in collaborative filtering recommender systems," unpublished

[9] Belkacem Chikhaoui, Mauricio Chiazzaro, Shengrui Wang, " An improved hybrid recommender system by combining predictions," in Workshops of International Conference on Advanced Information Networking and Applications, 2011.