A Hybrid Approach for Personalized Recommender System Using Weighted TFIDF on RSS Contents

Rebecca A. Okaka
SCIT, JKUAT
Nairobi, Kenya

Waweru Mwangi
SCIT, JKUAT
Nairobi, Kenya

George Okeyo
SCIT, JKUAT
Nairobi, Kenya

Abstract: Recommender systems are gaining a great popularity with the emergence of e-commerce and social media on the internet. These recommender systems enable users' access products or services that they would otherwise not be aware of due to the wealth of information on the internet. Two traditional methods used to develop recommender systems are content-based and collaborative filtering. While both methods have their strengths, they also have weaknesses; such as sparsity, new item and new user problem that leads to poor recommendation quality. Some of these weaknesses can be overcome by combining two or more methods to form a hybrid recommender system. This paper deals with issues related to the design and evaluation of a personalized hybrid recommender system that combines content-based and collaborative filtering methods to improve the precision of recommendation. Experiments done using MovieLens dataset shows the personalized hybrid recommender system outperforms the two traditional methods implemented separately.

Keywords: recommender systems; collaborative filtering; content-based filtering; hybrid recommender system; vector space model; term frequency inverse document frequency.

1. INTRODUCTION

Changes in information seeking behavior can be observed globally [1]. Rapid increase in blogs and websites has led to an increase in information overload and it has become extremely difficult for users to locate current relevant information, with vague ideas on where to get information, users often get lost or feel uncertain when seeking information on their own, giving rise to the need for creating systems that are able to process the existing information on one side, and help users by suggesting products, services or articles that match their tastes and preferences on the other side. Recommender systems (RS) are promising tools to deal with these issues.

There are lots of taxonomies of RS. They can be divided according to the fact whether the created recommendation is personalized or non-personalized [2]. Some Research distinguishes three main categories of personalized RS: collaborative filtering (CF), content-based filtering (CBF), and hybrid filtering (HF) [3]. Adomavicius and Tuzhilin claim that these three categories are the most popular and significant recommendation methods. However, they pinpoint the shortcomings of these methods when used individually such as limited content analysis, new item problem, new user problem, sparsity, scalability etc, which leads to poor recommendation quality. They also propose possible improvements; such as combining two or more recommender filtering methods using different hybridization techniques to overcome the challenges of single recommender systems.

In CF, a user gets recommendations of items that he or she hasn't rated or liked before, but that were

already positively rated by users in his or her neighborhood. In CBF, a user gets recommendations of items he or she had not seen or rated but similar to the ones he or she had rated or liked earlier. HF combines two or more filtering methods to overcome the limitations of each method. According to Tuzhilin et al, [4] the combination of two or more filtering methods proceeds in different ways; creating a unified model recommender system that brings all approaches together, utilizing some rules of one approach into a different approach and vice versa, separate implementation of algorithms and then joining results, developing one model that applies the characteristics of both methods.

The hybrid approach presented in this paper uses the weighted hybridization technique which probably is the most straight forward architecture for a hybrid system. Weighted hybridization technique was successfully used by the winners of the Netflix Prize competition [5]. Our approach involves separate implementation of algorithms then joining results, it is based on the idea of merging predicted ratings computed by individual recommenders to form a ranked list of items from which top (top k, k=5) items are selected and presented to the user as recommendations.

This hybrid approach combines CBF and CF methods, while CBF are able to make predictions on any item, CF only score an item if there are peer users who have rated it, the combination of these two methods therefore also helps eliminate the new item problem in CF and new user problem in CBF. This hybrid approach adapts the Vector Space Model (VSM) in both CBF and CF, uses ranking algorithm Term Frequency Inverse Document Frequency (TFIDF) and cosine similarity measure to find the relationships among users U, items I and attributes A.

Generally, in a recommender system, there exists a large number of m items $I= \{i_1, i_2....i_m\}$, which are described by a set of l attributes, $A= \{a_1, a_2....a_l\}$, where each item is described by one attribute or more, a number of n users, $U= \{u_1, u_2....u_n\}$ and for each user u, a set of rated items $IR_u = \{u_{i1}, u_{i2}, ..., u_{in}\}$. For $u \; \epsilon \; U$ and $i \; \epsilon \; I$, the recommender system predicts the rating $r'_{u,i}$ called

the predicted rating of the user u on the item i such that $r'_{u,i}$ is unknown. From this formulation, the main problem is predicting the rating a user would give an item he or she have not seen, then computing the accuracy of the predicted rating.

The main contribution of this work is that it provides a very straight forward hybrid architecture that can be used to improve recommendation precision as well as provide top most relevant items to users as recommendations. Because of the two methods used; content-based and collaborative filtering, the new user and new item problems is eliminated; the new user problem in content-based filtering is eliminated by collaborative filtering and the new item problem in collaborative filtering is eliminated by content-based filtering. This hybrid approach uses the most widely used effective information retrieval model, the VSM, and a very simple efficient ranking algorithm tfidf.

The rest of this paper is organized as follows; section 2 reviews related work. Section 3 presents the hybrid model and experimental results are presented in section 4. Section 5 presents conclusions and outlines of future research.

2. RELATED WORK

Hybrid recommender systems combine two or more recommender systems. Depending on the hybridization approach different types of systems can be found [6]. There have been some works on using boosting algorithms for hybrid recommendations [7, 8]. These works attempt to generate new synthetic ratings in order to improve recommendation quality. The personalized hybrid recommender system combines collaborative and content-based information.

Spiegel [9] proposed a framework that combines CBF, CF and demographic information for recommending information sources such as web pages or news articles. The author used home HTML pages to gather demographic information of users. The recommender system is tested on very few numbers of users and items which cannot guarantee the efficiency of the proposed system.

The author does not also give an explanation on how the model is built.

Melville [10] proposed a model in which content-based algorithm is used to enhance the existing user data then the collaborative filtering is used for rating prediction. But fails to justify how both approaches combined improves prediction accuracy. Another researcher [11] used a number of collaborative filtering algorithms such as Singular Value Decomposition (SVD), Asymmetric Factor Model and neighborhood based approaches to build a recommender system. The author shows that linearly combining these algorithms increases the accuracy of prediction, but the use of all these models leads to significant increase in training time.

Basu et al. [12] use Ripper, a rule induction system, to learn a function that takes a user and movie and predicts whether the movie will be liked or disliked. They combine collaborative and content information, by creating features such as comedies liked by user and users who liked movies of genre X. They however do not show how that approach improved recommendation quality.

Several other hybrid approaches are based on traditional CF, but also maintain a content-based profile for each user. These content-based profiles, rather than co-rated items, are used to find similar users. In Pazzani's approach [13], each user profile is represented by a vector of weighted words derived from positive training examples using the Winnow algorithm. Predictions are made by applying CF directly to the matrix of user profiles as opposed to the user ratings matrix. An alternative approach by; Fab [14] uses relevance feedback to simultaneously mold a personal filter along with a communal topic filter. Documents are initially ranked by the topic filter and then sent to a user's personal filter. The user's relevance feedback is used to modify both the personal filter and the originating topic filter. Good et al. [15] use collaborative filtering along with a number of personalized information filtering agents. Predictions for a user are made by applying CF on the set of other users and the active user's personalized agents.

The proposed hybrid approach adapts some interesting features of the above systems; the use of collaborative and content information. It however uses the VSM, tfidf and cosine similarity measure which are very simple efficient algorithms that enable item ranking based on weights. Prediction accuracy is computed by getting the deviation of the predicted rating from the actual rating. Other works on hybrid recommender systems can be found in [16].

# 3. THE HYBRID FILTERING MODEL
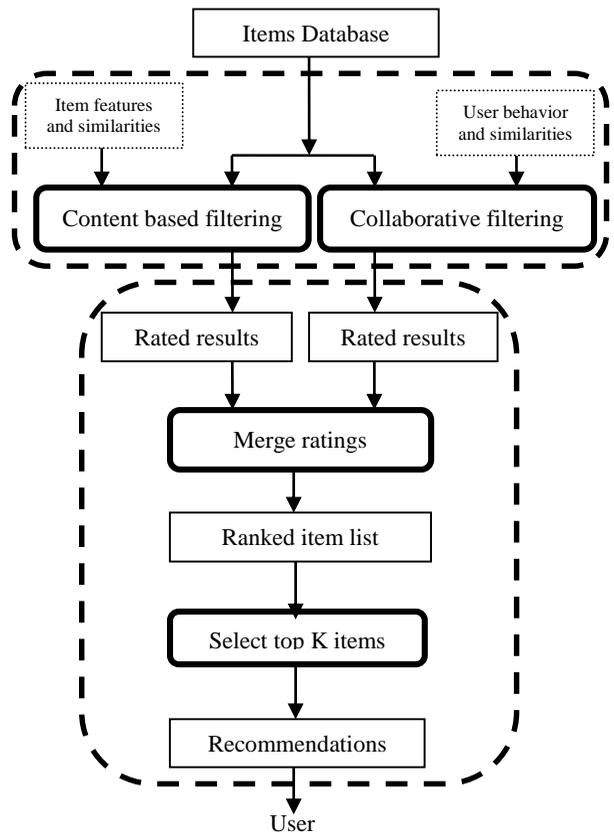
Figure 1. The hybrid filtering model

available on different domains, the model implements both CBF and CF methods separately. The two methods used (CBF and CF) complement each other and contribute to each other's effectiveness [17]. This hybrid approach uses the VSM on CBF and CF methods, tfidf and cosine similarity measure to compute relationships among items and users. CF and CBF methods are used to obtain separate ratings or score for every item. The more than one rating for every item are merged into a single value. The items are then ranked to form a single list of ranked items based on their scores, a set of items (e.g. top K, where K equals 5, 10….) topping the list (with highest scores or ratings) are finally presented to the user as recommendations.

3.1. The Vector Space Model

The vector space model [18] (VSM) is a standard algebraic model commonly used in information retrieval (IR). It treats a textual document as a bag of words, disregarding grammar and even word order. It represents both documents and queries by term sets and compares global similarities between documents and queries. The VSM typically uses tfidf (or a variant weighting scheme) to weight the terms. Then each document is represented as a vector of tfidf weights. Queries are also considered as documents. Cosine similarity is used to compute similarity between document vectors and the query vector. The term frequency $TF_{t,d}$ of term t in document d is defined as the number of times that a term t occurs in a document d. Note that;

$$TF_{t,d} = 1 \text{ if t exists in d} \qquad (1)$$

$$TF_{t,d} = 0 \text{ if t does not exist in d} \qquad (2)$$

It positively contributes to the relevance of d to t. The inverse document frequency IDF_t of term t measures the rarity of t in a given corpus. If t is rare, then the documents containing t are more relevant to t. IDF_t is obtained by dividing N by DF_t and then taking the logarithm of that quotient, where N is the total number of documents and DF_t is the document frequency of t or the number of documents containing t. Formally;

$$IDF_t = \frac{\log_{10} N}{DF_t} \qquad (3)$$

The TFIDF value of a term is commonly defined as the product of its TF and IDF values.

$$TF\text{-}IDF_{t,d} = TF_{t,d} \times IDF_t \qquad (4)$$

The TF-IDF weight W, for each term in a document d is given by;

$$W_{t,d} = (1 + \log_{10} TF_{t,d}) \times \frac{\log_{10} N}{DF_t} \qquad (5)$$

Generally;

$$W_{t,d} = \frac{1 + \log_{10} N}{DF_t} \text{ if } TF_{t,d} > 0 \qquad (6)$$

$$W_{t,d} = 0, \text{ otherwise} \qquad (7)$$

3.1.1 The Vector Space Model in Content-based filtering

Suppose a user profile is denoted by U and item profiles by I. $TF_{i,j}$ is the number of times the term t_i occurs in item $I_j \in I$, and the inverse document frequency of a term $t_i \in I_j \in I$ is calculated as;

$$IDF_i = \log_{10} I / DF_i \qquad (8)$$

Where DF_i is equal to the number of items containing t_i and I is equal to the total number of items being considered. Therefore;

$$TFIDF = TF_{i,j} \times IDF_i \qquad (9)$$

The TFIDF of each term is then calculated, and the vector of each user profile and item profiles are constructed based on their included terms. These vectors have the same length, so the similarity of these profiles can be calculated as;

$$Sim(U,I) = \frac{U \cdot I}{|U| \times |I|} = \frac{\sum_1^t tfidf_U \times tfidf_I}{\sqrt{\sum_1^t tfidf_U^2 + \sum_1^t tfidf_I^2}} \qquad (10)$$

The resulting similarity should range between from 0 to 1. If Sim(U,I)=0, then the two profiles are independent and if Sim(U,I) > 0, the profiles have some similarity. Information about a set of items

with similar rating patterns compared to the item under consideration is the basis for predicting the rating a U_i would give the item. The prediction formula is;

$$\text{Pred}(U_{i,}\, I_a) = \frac{\sum \text{similarity}(U_i, I_b) \times r_{Ui,Ia}}{\sum \text{similarity}(U_i, I_b)} \qquad (11)$$

Normally, the predicted rating of a user *u* for an item *i* in CBF is the average rating of the user on items viewed, therefore equation 11 can also be written as;

$$r'_{u,i}|\text{CBF} = \frac{\sum \text{similarity}(U_i, I_b) \times r_{Ui,Ia}}{\sum \text{similarity}(U_i, I_b)} \qquad (12)$$

$$r'_{u,i}|\text{CBF} = r_{Ui,Ia} \qquad (13)$$

Where $r_{Ui,Ia}$, isthe average rating of Ui on items already is viewed, and $r'_{u,i}|$CBFis the predicted rating of a user on an item in CBF.

3.1.2 The Vector Space Model in Collaborative filtering

The user profiles are represented as both documents and queries in an n-dimensional matrix. The weight for each term t in a user profile p is given by:

$W_{i,\,j} = TF_{i,j} \times IDF_i$ which can also be written as;

$$W_{i,\,j} = TF_{i,j} \times \log_{10} P\, /\, p_i \qquad (14)$$

$$IDF_i = \log_{10} P\, /\, p_i \qquad (15)$$

Where, $TF_{i,j}$ is the frequency of a term t in a profile p, P is the total number of profile, p_i is the total number of profiles containing term t and $W_{i,\,j}$ is the weight of the ith term in a profile j. The similarity between user U_i and user U_j is calculated using cosine similarity measure. The equation for calculating the similarity is as follows;

$$\text{Sim}(U_i, U_j) = \frac{U_i \cdot U_j}{|U_i| \times |U_j|} = \frac{\sum_{k=1}^{n} \text{tfidf}_{k,i} \times \text{tfidf}_{k,j}}{\sqrt{\sum_{k=1}^{n} \text{tfidf}_{k,i}^2 + \sum_{k=1}^{n} \text{tfidf}_{k,j}^2}} (16)$$

Again the resulting similarity should range between from 0 to 1. If $\text{Sim}(U_i, U_j) = 0$, then the two users are independent and if $\text{Sim}(U_i, U_j) = 1$, the users are similar. The information about a set of users with a similar rating behavior compared to the current user is the basis for predicting the rating a user U_i would give an item he or she has not rated. Based on the nearest neighbor of user U_i it is easy to determine the prediction of user U_i.

$$\text{Pred}(U_i, I) = r'_i + \frac{\sum \text{similarity}(U_i, U_j) \times (r_{j,item} - r'_j)}{\sum \text{similarity}(U_i, U_j)} \quad (17)$$

Where, U_j is U_i nearest neighbor, r'_i is the average rating of U_i, $r_{j,item}$ is the rating of U_j on the given item and r'_j is the average rating of U_j. Also, given that the predicted rating of a user u on an item I in CF is given as $r'_{u,i}|$CF, equation 17 can therefore be written as:

$$r'_{u,i}|\text{CF} = r'_i + \frac{\sum \text{similarity}(U_i, U_j) \times (r_{j,item} - r'_j)}{\sum \text{similarity}(U_i, U_j)} \qquad (18)$$

3.2 Hybridization Process

Table1. Extended user-item, user-user matrix

		Item					User profile-Attribute tf-idf					User-User cosine similarity				
		i_1	i_2	i_3	…	i_m	a_1	a_2	a_3	…	a_l	u_1	u_2	u_3	…	u_n
User	u_1	-	-	4	…	3	0.04	0	0	…	0	1	0.1	0	…	0.2
	u_2	4	2	-	…	5	0	0.01	0.02	…	0.02	0.1	1	0.1	…	0
	u_3	-	-	3	…	-	0.04	0	0	…	0.02	0	0.1	1	…	0
	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
	u_n	-	3	-	…	-	0.04	0.01	0.02	…	0.02	0.2	0	0	…	1
Item-Attribute tfiidf	a_1	0.0	0.0	0.0	…	0.0										
	a_2	0.0	0.01	0.0	…	0.01										
	a_3	0.02	0.02	0.0	…	0.0										
	…	…	…	…	…	…										
	a_l	0.02	0.0	0.0	…	0.0										
User-Item cosine similarity	u_1	0.3	0.2	0.1	…	0.1										
	u_2	0.3	0.0	0.0	…	0.0										
	u_3	0.1	0.5	0.3	…	0.4										
	…	…	…	…	…	…										
	u_n	0.0	0.2	0.4	…	0.2										

Rated item

Unrated item

As stated earlier the HF model combines CBF and CF which uses user-item matrix and user-user matrix respectively. Table 1 shows the model matrix with sample tfidf and cosine similarity scores among users and items. This model is based on the idea of deriving recommendation items by combining predictions computed by each individual recommenders CBF (Eq. 13) and CF (Eq. 18), here the separate scores of an individual recommender on an item $i \,\epsilon\, I$ recommended to a user $u \,\epsilon\, U$ are merged into a single unit.

To take into account the difference in the contribution of each predictor in the final rating prediction, each predictor is assigned a parameter. Such that the resulting rating prediction $r'_{u,i}|\text{HF}$ of a user u on an item i from HF is computed as follows;

$$r'_{u,i}|\text{HF} = \mu r'_{u,i}|\text{CBF} + \alpha r'_{u,i}|\text{CF} \qquad (19)$$

Where $\mu r'_{u,i}|\text{CBF}$ and $\alpha r'_{u,i}|\text{CF}$ are the predicted rating of an item $i \,\epsilon\, I$ for user $u \,\epsilon\, U$ in CBF and CF respectively.

To compute the value for each parameter, a function $S(n)$ that gives the weight of a user's rating n ($n=|IR_u|$) is used. The sigmoid function satisfies these constraints for $S(n)$.

The parameters μ and α can be computed using the sigmoid function as follows;

$$\mu = \frac{1}{1+e^{-n}} \qquad (20)$$

$$\alpha = 1 - \frac{1}{1+e^{-n}} \qquad (21)$$

These parameters μ and α, represent the weight confidence levels given to CBF and CF respectively. The resulting rating predictions of items from the hybrid approach are ranked based on their prediction scores, from the ranked items list the top scoring set of items (top k items) are selected and provided to the user as recommendations.

4. HYBRID DESIGN
4.1 System Physical Architecture
Figure 2 below shows the physical architecture of the proposed hybrid recommender system; it shows a set of simpler systems each with its own local context that is independent but not inconsistent with the context of the larger system as a whole. Both servers could still be physically implemented in a single network node.
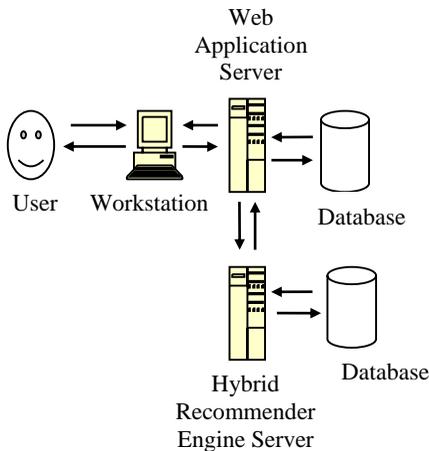
Figure 2. The Physical architecture

4.2 System Component Diagram
Figure 3 shows a simple component viewpoint of the Hybrid Recommender system. The Hybrid Recommender module, while calculating the accurate recommendation, uses the data stored in the Database module via a RESTful API

(Representational Sate Transfer Application Program Interface). The Hybrid Recommender module executes the methods on the background. It is connected to the User Interface module via the RecommendationRetrieval interface that enables the resulting recommendations to be shown to the user.

Figure 3. The Component Diagram

4.3 System Activity Diagram
The following activity diagram shows the flow of events within the proposed hybrid approach. It shows how the user interacts with the system.
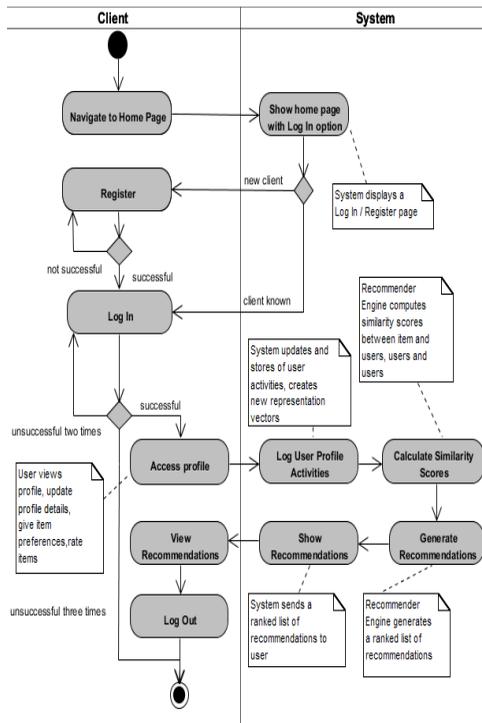
Figure 4. The Activity Diagram

5. EXPERIMENTS AND RESULTS

5.1 Dataset

The MovieLens (http://www.grouplense.org) 100k dataset was used. This data was collected by the GroupLens Research Project at the University of Minnesota during a seven-month period between 19th September 1997 and 22nd April 1998. The MovieLens is used mainly because it is publicly available and has been used in many hybrid recommender systems and therefore considered a good benchmark for this purpose. This dataset contains 943 users, 1682 movie items and 100000 ratings. Each user rates a minimum of 20 movies using integer values 1 to 5 and not all movies are rated by all users. There are 19 movie genres. A movie can belong to more than one genre. A binary value of 1 and 0 is used to indicate whether a movie belongs to a specific genre or not. The dataset is split into 5 subsets, each having (80%) training and (20%) test sets.

5.2 Evaluation metrics

The evaluation was done using prediction accuracy metric: Mean Absolute Error (MAE), which is used to represent how accurately a RS estimates a user's preference for an item. MAE is calculated by averaging the absolute deviation of a user's predicted score and actual score. The smaller the MAE the more precise the RS.

$$MAE = \frac{\sum_i^n |s_i - p_i|}{n} \qquad (22)$$

Where, n is the total number of items, i is the current item, s_i is the actual score a user expressed for item i, and p_i is the RS's predicted score a user has for i.

In this experiment 5-fold cross validation was performed on sub datasets 1 to 5 provided by MovieLens 100k dataset, 80% training data and 20% test data on each sub dataset. This experiment compares the results of the hybrid approach to CBF and CF methods implemented separately.

5.3 Results

Even though the 5 sub data sets used have almost the same number of users and items, they have different rating patterns therefore a standard number of users and items were used for experiment across all the datasets. Results presented here are the average of MAE across all the sub data sets given the specified number of users and items.

.

Table 2. Average MAE given 100 items

No of Users	Filtering Methods		
	CF	CBF	HF
100	0.3686	0.3828	0.3433
350	0.3374	0.3632	0.3162
500	0.3398	0.3659	0.3161
800	0.3258	0.3555	0.3081
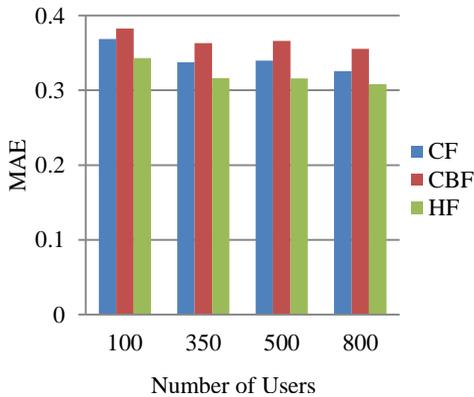
Figure 5. MAE given 100 items

Table 3. MAE given 500 items

No of Users	Filtering Methods		
	CF	CBF	HF
100	0.3396	0.3588	0.3043
350	0.3110	0.3560	0.2998
500	0.3016	0.3544	0.2954
800	0.2971	0.3519	0.2953
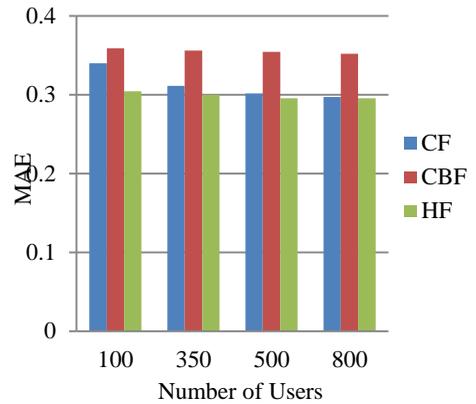
Figure 6. MAE given 500 items

Table 4. MAE given 700 items

No of Users	Filtering Methods		
	CF	CBF	HF
100	0.3326	0.3554	0.3029
350	0.3324	0.3690	0.3167
500	0.3203	0.3676	0.3122
800	0.3020	0.3564	0.2986
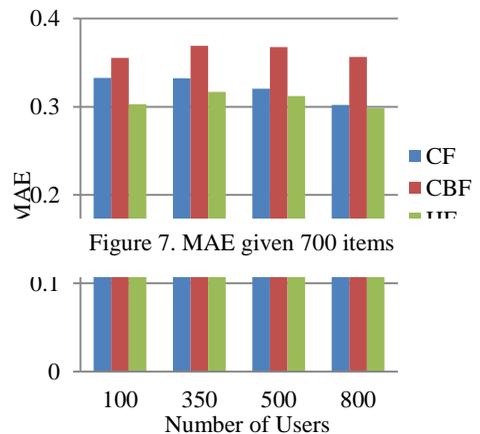
Figure 7. MAE given 700 items

Table 5. MAE given 1200 items

| No of Users	Filtering Methods		
	CF	CBF	HF
100	0.3374	0.3539	0.305
350	0.3386	0.3859	0.3245
500	0.3294	0.3745	0.3145
800	0.2997	0.3442	0.2883
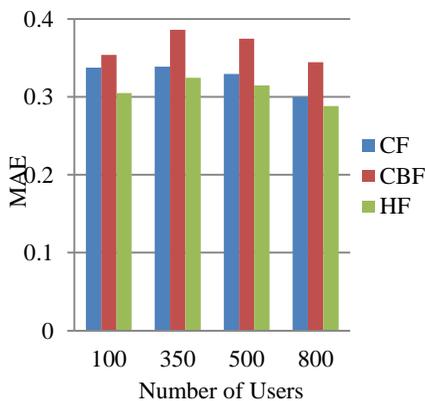
Figure 8. MAE given 1200 items

Collaborative filtering contributes greatly in the results of this approach more so where there are large numbers of items; its performance becomes better with increasing number or items and users respectively, but does not perform as well with large number of users and small number of items. On the other hand, content-based filtering does not make much contribution to this approach, its performance worsens as the number of items increases and its prediction is worse in cases where there are small number of users and items respectively.

However, across all of the evaluations, results show that the hybrid filtering model achieves better prediction accuracy than each of the traditional filtering methods implemented separately.

Collaborative filtering and content-based filtering performing on average 6% and 17% worse than the hybrid approach respectively; the hybrid approach achieves an average MAE of 0.3084 whereas collaborative and content-based filtering achieve 0.3258 and 0.3622 respectively.

6. CONCLUSIONS AND FURTHER WORK

In this paper a hybrid approach that combines content-based and collaborative filtering methods has been used to improve recommendation accuracy. Both methods use the effective information retrieval model the VSM, a very simple efficient ranking algorithm TFIDF and cosine similarity measure to find the relationships among users, items and attributes. The evaluation of the proposed hybrid model using real data has proven it achieves better prediction accuracy compared to a single content-based and single collaborative based recommender system. Because of this good performance, this hybrid recommendation approach and the information retrieval methods can therefore be adapted in different domains for recommendation purposes.

The possible future work related to this study is first to test the efficiency of this approach to other larger datasets and secondly, to explore the possibilities of experimenting with other variants of tfidf, similarity measures and the vector space model to see how well they perform in this kind of hybrid recommender environment.

7. REFERENCES

[1] Gavgani V.Z. "Health Information Need and Seeking Behavior of Patients in Developing Countries' Context; an Iranian Experience," Proceedings of the 1st ACM International Health Informatics Symposium, 2010, paper 11–12, pp. 575–579.

[2] Kazienko P., Kołodziejski P. 2005. 'WindOwls – Adaptive Systems for the Integration of Recommendation Methods in E – commerce', Springer Verlag, 218 – 224.

[3] Adomavicius G., Tuzhilin A. 2005. Toward the next generation of recommender systems:

A survey of the state-of-the-art and possible extensions. IEEE Trans. Knowl. Data Eng. (2005); 17:734-749.

[4] Tuzhilin A., Adomavicius G., 2005. Towards the next generation of recommender systems. A survey of the state of the art and possible extensions. IEEE Trans Knowl Data Eng, 17:734 – 749.

[5] Bell R., Koren Y., and Volinsky Ch. Chasing $1,000,000: How we won the Netflix Progress Prize. ASA Statistical and Computing Graphics Newsletter, 18(2):4–12, 2007.

[6] Burke R. Hybrid recommender systems: survey and experiments, User Modeling and User Adapted Interaction 12 (4) (2002) 331 – 370.

[7] Melville P., Mooney R., Nagarajan R. Content-boosted collaborative filtering for improved recommendations, in:18th National Conference on Artificial Intelligence (AAAI-02), 2002, PP. 187-192.

[8] Park S. T., Pennock D., Madani O., Good N., DeCoste D. Naïve filterbots for robust cold start recommendations, in: KDD '06: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Minning, 2006, pp. 669-705.

[9] Spiegel S., Kunegis J., Li F. "Hydra: a hybris recommender system [cross-linked rating and content information] in CIKM-CNIKM, 2009, pp. 75-80.

[10] Pazzani M. J. "A framework for collaborative, content based and demographic filtering," Artfi.Intell. Rev., vol. 13, no. 5-6, 1999, pp. 393-408.

[11] Melville P., Mooney R. J. Nagarajan R. "Content boosted collaborative filtering for improved recommendation," in proceedings of AAAI/IAAI, 2002, pp.187 – 193.

[12] Basu C., Hirsh H., Cohen C. Recommendation as classification: Using social and content-based information in recommendation. In Proceedingsof the Fifteenth National Conference on Artificial Intelligence (AAAI-98), 1998, pp 714–720.

[13] Pazzani A., Michael J. A framework for collaborative, content-based and demographic filtering. Artificial Intelligence Review, 1999, 13(5-6):393–408.

[14] Marko B., Yoav S. Fab: Content-based, collaborative recommendation. Communications of the Association for Computing Machinery, 1997,40(3):66–72.

[15] Good N., Schafer J. B., Konstan J.A., Borchers A., Sarwar B., Herlocker J., Riedl J. Combining collaborative filtering with personal agents for better recommendations. In Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI-99), 1999, pp 439–446.

[16] Jahrer M., Toscher A., Legenstein R. "Combining predictions for accurate recommender systems," in Proceedings of the SIGKDD conference. New York, NY, USA: ACM 2010, pp. 693-702.

[17] Burke R. D. "Hybrid recommender systems" survey and experiments," User Model, User-Adapt. Interact, vol 12, no. 4, 2002, pp. 331-370.

[18] Montaner, M., Lopez, B. and De la Rosa J.L.2003. 'A Taxonomy of Recommender Agents on the Internet', Artificial Intelligence Review, Kluwer Academic Publisher, 19, 285 – 330.