An Enhanced Content-Based Recommender System for Academic Social Networks

Vala Ali Rohani
Faculty of Computer Science and Information Technology,
University of Malaya
50603 Kuala Lumpur, Malaysia
vala@um.edu.my

Zarinah Mohd Kasirun
Faculty of Computer Science and Information Technology,
University of Malaya
50603 Kuala Lumpur, Malaysia
zarinahmk@um.edu.my

Kuru Ratnavelu
Institute of Mathematical Sciences,
Faculty of Science
University of Malaya
50603 Kuala Lumpur, Malaysia
kuru@um.edu.my

***Abstract*— the present study utilizes social computing techniques to enhance the content-based recommender systems. Coined as Enhanced Content-based Algorithm using Social Networking (ECSN), this recommender algorithm is applied in academic social networks to suggest the most relevant items to members of these online societies. In addition to considering user's own preferences, ECSN takes advantage of the interest and preferences of user's friends and faculty mates for providing more accurate recommendations. The research experiments were conducted by applying four different algorithms - random, collaborative, content-based, and ECSN, for 14 consecutive weeks. During this period, 1398 academic items were recommended to all 920 members of Malaysian Experts Academic Social Network (MyExpert). ANOVA tests indicate that the proposed algorithm significantly improves the prediction accuracy of algorithms based on well-known measurements of precision, fallout and F1. It is believed that this study can make a significant contribution to the level of user satisfaction in academic social networks.**

Keywords— Recommendation Algorithms; Prediction Accuracy; Academic Social Networks; Content-based recommender systems

I. INTRODUCTION

More than a billion users worldwide utilize online social networks for networking and accessing the information and updates according to their daily needs and preferences [1, 2]. However, the dramatic growth of online services and the vast variety of information available on the Web, have also led to some serious difficulties for users in making correct decisions over their online transactions. Users browsing websites expect to conveniently find highly relevant information based on their interests. Novel ways of communication and collaboration have emerged over the last few years [3]. Recommender Systems (RS) are software tools and techniques for suggesting the most related items to users [4, 5]. These suggestions may be in terms of different usages, such as relevant books to buy, persons who can be selected as friend in a social network, or interesting online news to be read. In other words, RSs provide a personalized recommendation based on information elicited from user profiles and items' specifications [6]. The primary goal of social recommender systems, thus, is to mitigate overloading experienced by social media users by recommending the most pertinent and attractive content. They also aim to raise the level of user engagement, adoption and participation through social media websites [7]. RSs may play a significant role in how social networks successfully ensure that their users receive suggestions with the most relevant and attractive content based on their preferences and interests [8].

RSs address the problem of information overloading by recommending the new and not-yet-experienced options to users according to their current needs. To provide these coherent suggestions, RSs need various types of information about users' preferences, the list of available items, and the history of previous interactions of the user with the system stored in designated databases [9, 10]. Browsing the recommended items, the user can then provide an implicit or explicit feedback that will be used for generating the more relevant recommendations in the future suggestions [10].

Bonhardet al. [11] point out that recommender systems, which are designed to alleviate the abundance of information, have so far not been very successful, especially in social networking domains. In their view, there is a need to integrate recommender systems and social networking functionality. It has been argued that recommender algorithms could be significantly improved by drawing on features from social systems [12]. Although recommender systems have been briefly studied in the past decade, the study of social-based recommender systems is a recent phenomenon [13]. As of now, the majority of recommender algorithms have focused on improving the performance of the recommendation process without considering the social elements of decision making and advice seeking [13]. More specifically, the traditional recommender systems have ignored social relationships among users. In our real life, for example, when we are asking our friends for recommendations of a nice restaurant, we are actually requesting verbal social recommendations [14]. Hence, in order to improve recommender systems and to provide more personalized

978-1-4799-6719-3/14 $31.00 © 2014 IEEE
DOI 10.1109/BDCloud.2014.131

recommendation results, social network information among users need to be incorporated [15].

In the present study, an enhanced version of content-based recommender systems is proposed, which takes advantage of social network based factors to improve the performance of recommendation process in academic social networks. While the pure content-based recommender algorithm considers only the given user's preferences, the proposed ECSN algorithm takes into account the preferences of users' friends and faculty mates too. By improving the prediction accuracy of the recommendation process the higher level of satisfaction is achieved.

II. RELATED WORKS

The most important objective of recommender systems is to estimate the ratings for the items that are new for a user [16]. Ultimately, after calculating the estimated rates for the yet unrated items, an ordered list of most related items can be prepared and suggested to the target user. A number of previous studies have revealed the contribution of recommender systems in education. A collaborative filtering method was used in a research to recommend documents that will either encourage the users to expand their knowledge of a given topic or reinforce the knowledge which they already have [17]. Usu et al.[18], developed an expert system for EFL English reading recommendation based on the opinions and domain knowledge of English teaching experts. Subsequently, they proposed a personalized mobile language learning system that included a reading material recommendation mechanism for guiding EFL students to read articles that meet their preferences [19]. In another study in this domain, the impact of collaborative recommender systems was examined on college students' use of an online forum for English learning [20].Adomavicius and Tuzhilin [16] reviewed the previous literatures and classified the recommender systems in three main categories of collaborative, content-based and hybrid filtering approaches. Collaborative algorithms make the recommendations based on the items that people with similar preferences and interests preferred previously, while the users' own preferences through previous interactions are considered by content-based methods in predicting the new items. Hybrid approaches are a combination of former two algorithms. As this study proposes an enhanced algorithm for content-based recommender systems, this approach is described further.

To come up with a judgment representing the user's level of interest to a specific item, RSs try to match up the preferences retrieved from a user profile against the attributes of that item. Content-based recommendation systems have been used in a variety of domains ranging from recommending web pages, news articles, restaurants, television programs, and items for sale. Although the details of various systems differ, content-based recommendation systems share in common a means for describing the items that may be recommended, a means for creating a profile of the user that describes the users' preferences, and a means of comparing items to the user profile to determine what to recommend. In doing so, all content-based RSs provide a mechanism for keeping the user profiles updated [22].

Items that are supposed to be recommended to the user are represented in two different categories - structured and unstructured [23]. In structured approach, each item is shown by the same set of attributes and the values of these attributes are usually stored in database [22]. There are also other cases that need unstructured data for presenting and storing the items. Unrestricted texts such as news articles are examples of unstructured data. In contrast of structured data, there are no attribute names with well-defined values for this paradigm. Hence, one common approach to dealing with free text fields is to convert them to a structured representation. Consequently, the methods for creating user profiles that deal with structured data need to differ somewhat from those methods for unstructured data converted to structured data [22].

In context of structured items, the machine learning algorithms may be used to create a user profile from structured data [22]. There are several such algorithms that have been studied in previous researches. Decision trees have been used extensively in use with structured data [24-27]. Some other researches [28-30] took advantage of nearest neighbor method which stores all of its training data, in memory. In order to classify a new, unlabeled item, the algorithm compares it to all stored items using a similarity function and determines the "nearest neighbor" or the k nearest neighbors. The methods that help users to incrementally refine queries based on previous interactions are commonly referred to relevance feedback algorithms. The main objective of this approach is to enable users to rate documents with respect to their information need [31]. Some techniques that learn linear decision boundaries are referred to as linear classifiers. This algorithm has been used in some researches for text classification tasks [32, 33]. As another algorithm in this domain, researchers have recognized Naïve Bayes as an exceptionally well-performing text classification algorithm and have frequently adopted the algorithm in recent work [34, 35]. When facing to small number of structured attributes, the performance, simplicity and understandability of decision trees for content-based models are all advantages [22]. Kim et al. [27] adopted this method for personalizing advertisements on web pages. To provide personalized advertisements in that study, a hierarchical tree data structure was considered for storing the personal preference scores of a customer for each product category.

III. METHODS AND MEASUREMENTS

To design this research experiments, online study is used which is recognized as the best method considering that it collects the real feedbacks from live interactions of users with the system [10]. To prepare this real runtime environment, MyExpert is designed and developed for this study. MyExpert presently has 920 members from 10 Universities in Malaysia as is the first Malaysia-specific academic social network. The research experiments have been done by applying four recommender algorithms of random, collaborative, content-based, and ECSN in 14 consecutive weeks from 7th September to 26th December 2012. The four different metrics (precision, recall, fallout, and F1) were used for measuring the prediction accuracy and comparing the performance of mentioned recommender algorithms.

Precision plays a great role in instances where some sets of best results are required out of several possible alternatives [36-38].This measurement is the share of top results that are relevant. In this study, the relevant items defined include academic items visited by users and that are rated with more than 2 stars. Recall is recognized as another metric for measuring usage prediction in recommender systems and other information retrieval domains. It determines the proportion of all relevant results included in the top results [39]. In studies where a fixed number of recommendations are suggested to each user (such as the current study in which the top 10 items are recommended to MyExpert users in every week of experiments) precision and recall can be computed at each recommendation list length N for each user. Then the average value of precision and recall can be computed for all users involved in the experiment [40]. To further assure the performance of the ECSN algorithm, two other metrics were applied to evaluating the accuracy prediction of all studied recommender systems in this research. Fallout, or the false positive rate [38], is measured as the ratio of selected (recommended) items that are not used (irrelevant) to the total number of unutilized items. It is the probability that an irrelevant (not used) item will be recommended to a user. According to this definition, a lower fallout rate indicates better recommender algorithm performance. To evaluate the overall performance of a recommender algorithm it makes sense to consider precision and recall together [39]. Various researches have pointed out that precision and recall are inversely related and dependent on the length of the result list returned to the user [41]. So under these circumstances, a vector of precision/recall pairs may describe recommender system performance. Several methods have been assessed to combine precision and recall into a single metric [40, 42]. One approach is the F1 metric which amalgamates precision and recall into a single value.

IV. THE PROPOSED ALGORITHM

In reference to the experimental method selected for this research - online study - having a pool of users for interacting with different recommender systems was essential. Answering this need, MyExpert academic social network was developed. MyExpert members click and rate the academic items that were sent to them by four different recommender algorithms that were implemented in this study. In proposed model, the ECSN recommender engine is responsible for collecting the relevance feedback from MyExpert users, generating and keeping updated the user profiles based on their elicited preferences during the interactions with the systems. ECSN recommender algorithm is applied to find top 10 academic items among 100 submitted ones in each week of study and shapes the weekly e-Newsletter for each member of MyExpert.

However, by just considering the user’s own preferences causes a known problem in domain of recommender systems, cold-start problem. The origin of this shortcoming can be traced to this phenomenon that there isn’t any previously recorded interaction for new users. This is a problem that some recommender algorithms suffer from it such as collaborative and content-based approaches [44, 45]. To solve this problem, in addition of considering the users’ own preferences, ECSN algorithm utilizes the ‘Friends Profile’ and ‘Faculty mates Profile’ too. In doing so, all transactions records of a given user’s friends are analyzed by ECSN recommender engine and the most interesting nodes in preference tree structure of academic items would be elicited. And then, the pointing value of elicited nodes will be updated in preference tree of the given user. The same process would be done regarding the faculty mates of the target user.

In ECSN algorithm, the users’ preference scores for each academic item category are stored in a tree structure based on following definition:

Definition 1: The preference tree PT(i) of user i is isomorphic to the hierarchy tree of item categories, and the set of nodes of the preference tree PT(i) is as follows:

$$PT(i) = \{ (UID, ICID, PS)\}$$

where UID,ICID, and PS are the user identifier, the item category identifier of the hierarchy tree, and the preference score, respectively.

Definition 2: The preference scores (PS) are defined as follows:

$$PS(i,j) = \propto_1 \times SelfClickScore(i,j) + \propto_2 \times SelfRankScore(i,j) + \propto_3 \times FacultyMatesScore(i,j) + \propto_4 \times FriendsScore(i,j)$$

where $PS(i,j)$ is the total preference score of user i for the academic item category node j. Each element of this definition is described in following:

$SelfClickScore(i,j)$ is the score related to clicks of given user i for the item category node j , which is specified by counting the number of clicks for given customer during the research experiments.

$SelfRankScore(i,j)$ is calculated by adopting the submitted rates of given user i to academic items which classified in category node j.

the value of $FacultyMatesScore(i,j)$ is calculated by considering the top 3 interesting item nodes of preference tree among the members who have registered in the same faculty that the given user i belongs to.

Last element, $FriendsScore(i,j)$, is dedicated to preferences of friends for given member i. The strategy that described above for calculating the $FacultyMatesScore(i,j)$ is adopted here for $FriendsScore(i,j)$ with this difference that it takes into account the top 3 item nodes which are mostly interesting among friends of given member.

As mentioned in previous researches [22], some weights might be assigned to each parameters of the formula for computing the preference scores. Accordingly, in Definition 2, the ($\propto_k$) represent relative weights for each element. As $SelfClickScore$ and $SelfRankScore$ are the most important personal elements that should be counted for given users i, the value of 5 is considered for $\propto_1$and $\propto_2$. Relatively, the weight of 3 has been considered for $\propto_3$ since $FacultyMatesScore$ is less significant than user's own preferences. And finally, $\propto_4$ is set by 1 as $FriendsScore$ has the lowest influence in Definition 2.

After calculating the preferences scores PS for each user i, the Definition 3 is applied for some non-leaf nodes of preference tree which their values are still 0.

Definition 3: The preference scores (PS) a non-leaf-level product category j are defined as follows:

$$PS(i,j) = Average\ (PS(i,k))$$

k ∈ {k|k is a child node of product category j}

(1) When the given user i rates the academic items related to category node j:

$$PS(i,j).SelfRankScore = PS(i,j).SelfRankScore + RV$$

(2) When the given user i clicks the academic items related to category node j:

$$PS(i,j).SelfClickScore = PS(i,j).SelfClickScore + 1$$

The above update procedure does not require the update of preference scores for all nodes of the tree, rather it requires only the update of the preference scores of nodes relate to visited and rated items.

After updating SelfRankScore and, the preference scores (PS) need to be updating by considering the Faculty mates and Friends of given user i:

$$PS(i,j).FacultyMatesScore = PS(i,j).FaculltyMatesScore + Avearge\ (FS)$$

where

$$FS \in Top\ 3\ Preferences\ scores\ of faculty\ mates\ for\ given\ user\ i\ \}$$

Similarly, the FriendsScore value is updated as:

$$PS(i,j).FriendsScore = PS(i,j).FriendsScore + Avearge\ (FS)$$

where

$$FS \in Top\ 3\ Preferences\ scores\ of friends\ for\ given\ user\ i\ \}$$

In each week of experiments, 100 academic items were submitted at MyExpert academic social network. Each studied recommender algorithms aimed to select top 10 items for each user and recommend it through an e-newsletter. For first three algorithms, the selection process was implemented based on recommender algorithms that were studied through the review of literature. In random recommender algorithm, the random 10 items were selected. Collaborative algorithm made predictions based on the items that people with similar preferences and interests preferred previously [23, 24]. For implementing the pure content-based recommendation, the preference treeapproach [22] was followed. And after all, an enhanced selection process was used in ECSN recommender algorithm which is illustrated in Figure 1.

```
for each u ∈ U
{
    Generating the ordered stack of item categories
    SIC ← 0
    SC ← 0
    While (SelectedItems < 10)
    {
    TopItemCat ← POP (ItemStack) ,
    TopItemsList ←
    FindNewItems(TopItemCat, PrioritizedCount(SCC))
    SIC−= count(TopItemsList)
    Adding TopItemsList to RecommendationList
    }
}
```

Figure 1: The ECSN Selection Process

For each user of MyExpert, the item categories get ordered based on PS value and stored in a stack data structure (ItemStack) such that the category with biggest PS is accessible at the top of the stack. To produce the recommendation list for user u, top most category is moved to TopItemCat using POP (ItemStack). Then, the new submitted items in MyExpert (100 items per week) are searched to find academic items using category ID of TopItemCat. The recommendation list that is supposed to be suggested to each user u includes 10 most relevant items. To have more items with highest PS value in this list, the PrioritizedCount array has been considered to identify the number of items that should be for each top scored item category:

$$int\ PrioritizedCount = \{3,2,2,1,1,1,1,1,1,1\}$$

Based on this identified priority, the highest scored category can have up to 3 items while two next highest ones come with at most 2 items in recommendation list. The others have the same value of one item. In the body of while loop, the ordered recommendation list (TopItemsList) is generated for each user u∈U.

V. EXPERIMENTS AND RESULTS

This section presents the experimental results and also the evaluation of the performance of proposed algorithm (ECSN) based on four previously mentioned measurements. During 14 weeks of online experiments carried out among 920 members of MyExpert academic social network, precision, recall, fallout, and F1 were used for evaluating the performance of each studied recommender algorithms. A complete view of the results based on prediction accuracy measurements is presented in Table 1.

Table 1: Complete View of Results based on Prediction Accuracy Measurements

Recommender Algorithm	Data Gathering Series No.	Precision		Recall		Fallout		F1	
		Avg Value in each series	Average	Avg Value in each series	Average	Avg Value in each series	Average	Avg Value in each series	Average
Random	Series #1	0.217241379		0.97164751		0.079655298		0.35509129	
	Series #2	0.225974026		0.995238095		0.07876572		0.36831924	
	Series #3	0.180392157	**0.1693**	0.971895425	**0.9756**	0.083231321	**0.0844**	0.30430305	**0.2883**
	Series #4	0.151785714		0.961309524		0.086121657		0.26217532	
	Series #5	0.175641026		0.993589744		0.083744789		0.29851271	
Collaborative	Series #6	0.191304348	**0.2066**	0.990942029	**0.9263**	0.082245475	**0.0810**	0.32069714	**0.3373**
	Series #7	0.228571429		0.932738095		0.078659818		0.36716702	
	Series #8	0.2		0.855144558		0.081821206		0.324181	
Content Based	Series #9	0.209850746	**0.2132**	0.941044776	**0.9175**	0.078631394	**0.0801**	0.34317441	**0.3460**
	Series #10	0.215280702		0.918910914		0.080502074		0.34883662	
	Series #11	0.214536		0.892632275		0.081030192		0.34593072	
ECSN	Series #12	0.225423729	**0.2477**	0.946166263	**0.9523**	0.079236005	**0.0770**	0.36410063	**0.3928**
	Series #13	0.245454545		0.967820599		0.077034544		0.39159455	
	Series #14	0.272340426		0.942907801		0.074596385		0.42261639	

The first two runs of experiments were considered as pretest stages that MyExpert members received the recommended items through weekly e-Newsletters and accordingly were not included in measurements. During these two weeks, they tried rating the academic items. After this pilot test, each recommender algorithms of Random, Collaborative, Content-based and ECSN was applied in three subsequent weeks. For each series of experiments, the average value of four different measurements (Precision, Recall, Fallout, and F1) was calculated and used for measuring the average value for each recommender algorithm. The one-way Analysis of Variance (ANOVA) test can be used for the case of a quantitative outcome with a categorical explanatory variable that has two or more levels of treatment [46]. As four different measurements (Precision, Recall, Fallout, and F1) were used in this study for computing the prediction accuracy of recommender algorithms, the four ANOVA test were run to examine if there were any between-group differences of means between studied recommender algorithms (Table 2). According to the results of LSD post hoc tests, the mean value of precision is significantly different between ECSN and three other algorithms (Table 2.a) while this variation is not clear in terms of Recall measurement (Table 2.b).
The differences are obvious also based on Fallout and F1 as shown in Table 2.c and Table 2.d. Consequently, in exception of Recall, the other measurements (Precision, Fallout, and F1) show the significant difference of prediction accuracy between four studied recommender algorithms. Figure 2 illustrates the mean value of prediction accuracy based on four different measurements. As shown in Figure 2(a), the precision value had an upward trend in the 14 weeks of experiments. It started at 0.169 with the random algorithm and steadily rose to 0.207 for the collaborative and 0.213 for the content-based approach. In the last stage of the experiments, the ECSN algorithm reached a peak of 0.248. The ECSN algorithm enhanced the precision value of other recommender algorithms by 32% (random, MD= 0.00741), 20% (collaborative, MD=0.00395) and 21% (content-based, MD=0.00310). The recall value comparison for all four recommender algorithms can be seen in Figure 2(b). The highest rate was attained by the random (0.976) and the ECSN algorithm (0.952), while the collaborative and content-based approaches had lower recall values, at 0.926 and 0.918 respectively. Although the ANOVA test results doesn't show any significant differences in case of recall metrics, but even based on this measurement, the contribution of the ECSN method is clear with improvement over the collaborative by 3% and content-based method by 4%.

Table 2: One-way ANOVA Tests for Examining the Difference of Means between Recommender Algorithms

a. ANOVA-Test on Precision Value

	Rnd	CL	CB	ECSN
Rnd	1			
CL	0.03735	1		
CB	0.04395*	0.00660	1	
ECSN	0.07847*	0.04111*	0.03452*	1

b. ANOVA-Test on Recall Value

	Rnd	CL	CB	ECSN
Rnd	1			
CL	0.4932	1		
CB	0.05807	0.00875	1	
ECSN	0.02330	0.02602	0.03477	1

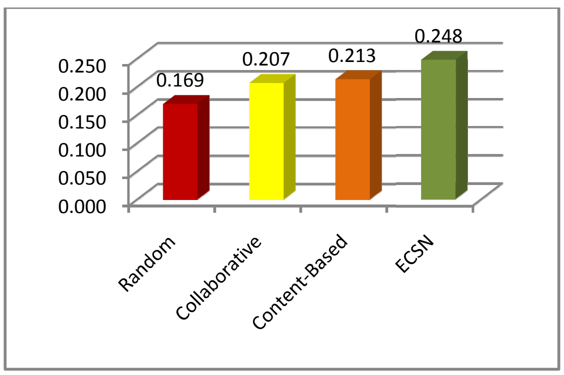
c. ANOVA-Test on Fallout Value

	Rnd	CL	CB	ECSN
Rnd	1			
CL	0.00346*	1		
CB	0.00431*	0.0085	1	
ECSN	0.00741*	0.00395*	0.00310	1

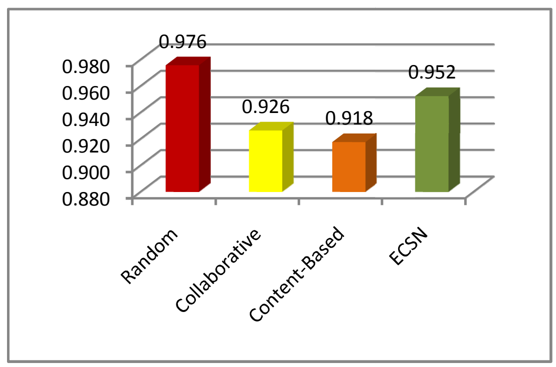
d. ANOVA-Test on F1 Value

	Rnd	CL	CB	ECSN
Rnd	1			
CL	0.04902*	1		
CB	0.05765*	0.00863	1	
ECSN	0.10444*	0.05542*	0.04679*	1

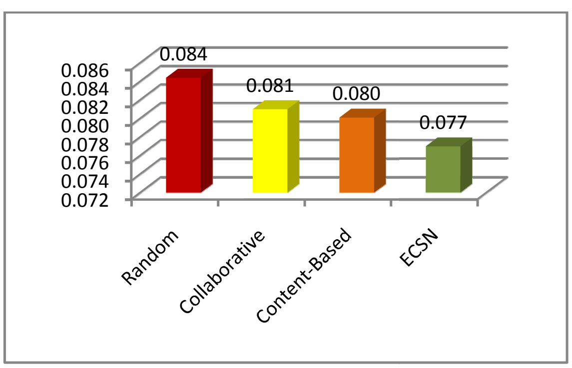
*the mean difference is significant at the 0.05 level.



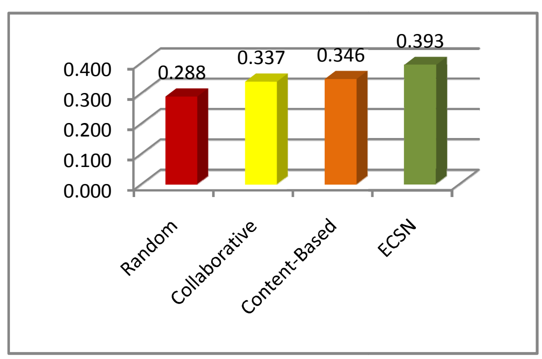
(a) Precision



(b) Recall



(c) Fallout



(d) F1

Figure 2: Comparison of Random, Collaborative, Content-Based, and ECSN Algorithms based on four Usage Prediction Metrics

The fallout values of the studied recommender algorithms are compared in Figure 2(c), where it is obvious that the fallout rate had a decreeing trend in the 14 weeks of experiments. Referring to the definition of fallout, a lower fallout rate indicates better recommender algorithm performance. Thus, this diagram shows that the prediction accuracy of recommender algorithms improved from random (0.084), to collaborative (0.081), content-based (0.080), and finally ECSN (0.077) algorithm while the values of the fallout metric declined steadily.

The final section of the diagram corresponds to the F1 score. As per Eq. 10, the values of both precision and recall are combined to calculate the F1 score for measuring the accuracy prediction of a given recommender algorithm. It is thus considered an overall metric that includes both recall and precision. Figure 2(d) shows the steady rise of F1 values during 14 weeks of experiments. The lowest value belongs to the random algorithm (0.288) while the peak of 0.393 corresponds to when the ECSN algorithm was applied. In other words, the ECSN recommender algorithm significantly contributes to the F1 values of the random (MD= 0.10444), collaborative (MD=0.05542), and content-based (MD=0.04679) algorithms, by 26%, 14%, and 12% respectively.

VI. CONCLUSION AND FUTURE DIRECTIONS

Although recommender systems have been briefly studied in the past decade, the study of social-based recommender systems is a recent phenomenon. The purpose of this research was to determine how the prediction accuracy of recommender systems in academic social networks could be improved by applying an enhanced content-based algorithm utilized by social networking features (ECSN). To test the effectiveness of the proposed recommender algorithm, a 14-week experiment was performed to compare the ECSN algorithm, Content-Based approach, collaborative filtering, and random method. The results of the experiment based on four different measurements indicate that ECSN recommender algorithm provides better overall prediction accuracy than other studied methods. In relation to precision, the ECSN algorithm had a significant contribution of 32% (random), 17% (collaborative), and 14% (content-based). In terms of Recall measurement, the contribution of the ECSN method is evident with improvement over the collaborative by 3% and content-based method by 4%. The fallout rate had a decreeing trend in the 14 weeks of experiments. Referring to the definition of fallout, a lower fallout rate indicates better recommender algorithm performance. Thus, based on this measurement, the prediction accuracy of recommender algorithms improved from random (0.084), to collaborative (0.081), content-based (0.080), and finally ECSN (0.077) algorithm while the values of the fallout metric declined steadily. F1, as an overall metric that includes both recall and precision, is the last measurement that was used in this research. The experimental results show the steady rise of F1 values during 14 weeks of experiments. The lowest value belongs to the random algorithm (0.288) while the peak of 0.346 corresponds to when the ECSN algorithm was applied. In other words, it is clear that the ECSN recommender algorithm significantly contributes to the F1 values of the random, collaborative, and content-based algorithms, by 27%, 14%, and 11% respectively. To conclude, 14 weeks of evaluations based on the four most familiar metrics, namely precision, recall, fallout, and F1, demonstrate that the proposed recommender algorithm in this research (ECSN) successfully enhanced the prediction accuracy compared to the other studied and implemented recommender approaches.

As a novel topic for future researches, the social network based features were utilized in this study can be combined with SNA concepts to propose a new model for enhancing the organizational behaviors and recommendations. The results of such researches can be used, for example, in decision support systems (DSS). Another question that needs more extensive research is whether the weights that were considered for calculating the node scores in Definition 2 can be optimized. In this research, based on the degree of importance, weight of 5 was considered for user's own preferences, weight 3 for taking into account of faculty mates, and finally for applying the preferences of friends the weight of one was considered. Although applying these weights could make a significant contribution in solving the cold start problem and also in improving the prediction accuracy of recommendations, it may be better to apply fussy logic or neural networks techniques to achieve even more optimum weights. Furthermore, MyExpert academic social network, which was developed in this study as the runtime environment for establishing the online experiments, has this potential to be used in future researches. Anomaly detection and community studies are two fields of related researches that can use this run time environment for establishing their experiments and archiving online results.

ACKNOWLEDGMENTS

The authors would like to thank the reviewers for their comments. This work was supported under Grants PS063-2010A.

REFERENCES

[1] Cheung, C.M.K., P.Y. Chiu, and M.K.O. Lee, Online social networks: Why do students use facebook? Computers in Human Behavior, 2011. 27(4): p. 1337-1343.

[2] Rohani, V.A. and O.S. Hock, On Social Network Web Sites: Definition, Features, Architectures and Analysis Tools. Journal of Advances in Computer Research, 2010. 1(2): p. 41-53.

[3] Lin, J.-W. and Y.-C. Lai, Online Formative Assessments with Social Network Awareness. Computers & Education, 2013.

[4] Burke, R., Hybrid web recommender systems. The adaptive web, 2007: p. 377-408.

[5] Mahmood, T. and F. Ricci. Improving recommender systems with adaptive conversational strategies. in the 20th ACM conference on Hypertext and hypermedia 2009. New York, NY, USA: ACM.

[6] Cacheda, F., et al., Comparison of collaborative filtering algorithms: Limitations of current techniques and proposals for scalable, high-performance recommender systems. ACM Transactions on the Web (TWEB), 2011. 5(1): p. 2.

[7] Guy, I. and D. Carmel. Social recommender systems. in Proceedings of the 20th international conference companion on World wide web. 2011. ACM.

[8] Guy, I., et al. Social media recommendation based on people and tags. in Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval. 2010. ACM.

[9] Afzal, M.T. and H. Maurer, Expertise Recommender System for Scientific Community. Journal of Universal Computer Science, 2011. 17(11): p. 1529-1549.

[10] Ricci, F., L. Rokach, and B. Shapira, Introduction to recommender systems handbook. Recommender Systems Handbook, 2011: p. 1-35.

[11] Bonhard, P., M.A. Sasse, and C. Harries. The devil you know knows best: how online recommendations can benefit from social networking. in Proceedings of the 21st British HCI Group Annual Conference on People and Computers: HCI... but not as we know it-Volume 1. 2007. New York, NY, USA: British Computer Society.

[12] Bonhard, P., et al. Accounting for taste: using profile similarity to improve recommender systems. in Proceedings of the SIGCHI conference on Human Factors in computing systems. 2006. York, NY, USA: ACM.

[13] Ma, H., et al. Recommender systems with social regularization. in Proceedings of the fourth ACM international conference on Web search and data mining. 2011. ACM.

[14] Bonhard, P. Who do trust? Combining recommender systems and social networking for better advice. in Proceedings of the Wokshop Beyond Personalization 2005, in conjunction with the International Conference on Intelligent User Interfaces IUI'05. 2005. San Diego, California.

[15] Zhou, X., et al., The state-of-the-art in personalized recommender systems for social networking. Artificial Intelligence Review, 2012. 37(2): p. 119-132.

[16] Adomavicius, G. and A. Tuzhilin, Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. Knowledge and Data Engineering, IEEE Transactions on, 2005. 17(6): p. 734-749.

[17] Mangina, E. and J. Kilbride, Evaluation of keyphrase extraction algorithm and tiling process for a document/resource recommender within e-learning environments. Computers & Education, 2008. 50(3): p. 807-820.

[18] Hsu, C.-K., G.-J. Hwang, and C.-K. Chang, Development of a reading material recommendation system based on a knowledge engineering approach. Computers & Education, 2010. 55(1): p. 76-83.

[19] Hsu, C.-K., G.-J. Hwang, and C.-K. Chang, A personalized recommendation-based mobile learning approach to improving the reading performance of EFL students. Computers & Education, 2012.

[20] Wang, P.-Y. and H.-C. Yang, Using collaborative filtering to support college students' use of online forum for English learning. Computers & Education, 2012.

[21] Mladenic, D., Text-learning and related intelligent agents: a survey. Intelligent Systems and their Applications, IEEE, 1999. 14(4): p. 44-54.

[22] Pazzani, M. and D. Billsus, Content-based recommendation systems. The adaptive web, 2007: p. 325-341.

[23] Smyth, B., Case-based recommendation, in The adaptive web2007, Springer. p. 342-376.

[24] Kim, J.W., et al., Application of decision-tree induction techniques to personalized advertisements on internet storefronts. International Journal of Electronic Commerce, 2001. 5: p. 45-62.

[25] Cohen, W.W. Fast effective rule induction. in Machine Learning International Workshop, 1995

[26] Cho, Y.H., J.K. Kim, and S.H. Kim, A personalized recommender system based on web usage mining and decision tree induction. Expert Systems with Applications, 2002. 23(3): p. 329-342.

[27] Kim, J.W., et al., A preference scoring technique for personalized advertisements on Internet storefronts. Mathematical and computer modelling, 2006. 44(1): p. 3-15.

[28] Allan, J., et al. Topic detection and tracking pilot study final report. in DARPA Broadcast News Transcription and Understanding Workshop. 1998. Lansdowne, Virginia.

[29] Yang, Y., An evaluation of statistical approaches to text categorization. Information retrieval, 1999. 1(1-2): p. 69-90.

[30] Billsus, D., M.J. Pazzani, and J. Chen. A learning agent for wireless news access. in Proceedings of the 5th international conference on Intelligent user interfaces. 2000. New York, NY, USA: ACM.

[31] Manning, C.D., P. Raghavan, and H. Schütze, Introduction to information retrieval. Vol. 1. 2008: Cambridge University Press Cambridge.

[32] Kivinen, J. and M.K. Warmuth, Exponentiated gradient versus gradient descent for linear predictors. Information and Computation, 1997. 132(1): p. 1-63.

[33] Zhang, T. and V.S. Iyengar, Recommender systems using linear classifiers. The Journal of Machine Learning Research, 2002. 2: p. 313-334.

[34] Nigam, K., et al. Learning to classify text from labeled and unlabeled documents. in Proceedings of the national conference on artificial intelligence. 1998. Citeseer.

[35] Yoshii, K., et al., An efficient hybrid music recommender system using an incrementally trainable probabilistic generative model. Audio, Speech, and Language Processing, IEEE Transactions on, 2008. 16(2): p. 435-447.

[36] Salton, G., Automatic Text Processing: The Transformation, Analysis, and Retrieval of information by computer1989: Addison-Wesley.

[37] Cremonesi, P., R. Turrin, and F. Airoldi. Hybrid algorithms for recommending new items. in Proceedings of the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems. 2011. ACM.

[38] Shani, G. and A. Gunawardana, Evaluating recommendation systems. Recommender Systems Handbook, 2011: p. 257-297.

[39] Herlocker, J.L., et al., Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems (TOIS), 2004. 22(1): p. 5-53.

[40] Sarwar, B., et al. Item-based collaborative filtering recommendation algorithms. in Proceedings of the 10th international conference on World Wide Web. 2001. ACM.

[41] Cleverdon, C., J. Mills, and M. Keen, Factors Determining The Performance Of Indexing Systems Volume 1. Design. 1966.

[42] Harman, D., Overview of the second text retrieval conference (TREC-2). Information Processing & Management, 1995. 31(3): p. 271-289.

[43] Perez, L.G., M. Barranco, and L. Martinez. Building user profiles for recommender systems from incomplete preference relations. in Fuzzy Systems Conference, 2007.

[44] Rohani, V. A., Kasirun, Z. M., Kumar, S., & Shamshirband, S. An effective recommender algorithm for cold-start problem in academic social networks. Mathematical Problems in Engineering, 2014.

[45] Zhang, Z.-K., et al., Solving the cold-start problem in recommender systems with social tags. EPL (Europhysics Letters), 2010. 92(2): p. 28002.

[46] Littell, R.C., SAS for mixed models2006: SAS institute.