Exploring Social Approach to Recommend Talks at Research Conferences

Danielle H. Lee
School of Information Sciences
University of Pittsburgh
Pittsburgh, PA., USA
hyl12@pitt.edu

Peter Brusilovsky
School of Information Sciences
University of Pittsburgh
Pittsburgh, PA., USA
peterb@pitt.edu

***Abstract*— This paper investigates various recommendation algorithms to recommend relevant talks to attendees of research conferences. We explored three sources of information to generate recommendations: users' preference about items (i.e. talks), users' social network and content of items. In order to find out what is the best recommendation approach, we explored a diverse set of algorithms from non-personalized community vote-based recommendations and collaborative filtering recommendations to hybrid recommendations such as social network-based recommendation boosted by content information of items. We found that social network-based recommendations fused with content information and non-personalized community vote-based recommendations performed the best. Moreover, for cold-start users who have insufficient number of items to express their preferences, the recommendations based on their social connections generated significantly better predictions than other approaches.**

Content-boosted Recommendation, Cold Start Problem, Social Networks, Social Network-based Recommendations, Hybrid Recommendation, ConferenceNavigator

I. INTRODUCTION

In research conferences, attendees get excited to learn current research trends, to find interesting papers, and to explore which talks and seminars they need to listen to in order to enhance their knowledge. More importantly, by attending talks, they expect to meet other researchers who are doing similar or relevant research with them and to share opinions about their topics, and in many cases, look for a chance of interesting research collaborations. However, they are also easily overwhelmed by the number of papers published in a conference and busy schedules of multiple sessions. In conferences, several sessions are typically held at a time, and a lot of research-related activities – such as tutorials, industrial discussion, social events, keynote speech, etc. – proceed for a short period of time. Thus, finding interesting talks to attend is a real challenge. There is little time for attendees to analyze all alternatives and decide where to go. It is easy for them to miss important talks and further to miss important opportunities of future collaborations. A recommender system that suggests attendees which talks are worthy to attend could be of real help in this context. Hence, we explore various recommendation algorithms to suggest talks which are worthy for each attendee to listen to – from pure collaborative filtering to hybrid approaches fusing collaborative filtering or social network-based recommendations with content information of talks. To our best knowledge, this is the first attempt to generate personalized recommendation for conference talks.

As an effective way to solve users' perceived information glut problem, recommendation technology has gained attentions not only from the academia, but also from industry and have succeed in various domains. However, there is no recommendation for conference talks which are very special occasions. The majority of recommendation algorithms were created for a large-scale and long-term context where user actions (such as ratings) are collected from many users over a long period of time. On the other hand, at conferences, the time to collect users' preference data is severely limited, usually right before the conferences and during the conferences. Since people who attend conferences are usually from a specific research community of the corresponding topics, users who express their preferences on the talks are also limited. Additionally, we suggest that this domain requires harmonizing information items (i.e. papers/talks) with social network context of users in recommendations. Research collaborations are rooted on overlapped research interests and topics between two people. Hence, when researchers attend conferences with their collaborators, talks caught their attentions could also interest their collaborators.

Therefore, in here, we consider various kinds of recommendations which utilize different set of inputs and explore what the best recommendation algorithm is for conference talks. Specifically, this study is based on an adaptive hypermedia system to support conference attendees' navigations and effective scheduling – Conference Navigator. This system was introduced in 2006 and evolved into the current version 3 (http://halley.exp.sis.pitt.edu/cn3). As of July 2012, 16 conferences have used this system.

In its essence, as Figure 1 shows, CN3 displays all talks in a conference with navigation supports. It also provides various information access methods from navigations via titles, presentation types, and authors' names to advanced search and personalized recommendations. Whenever users find interesting talks, they are able to bookmark them, and then the bookmarked items are added to their schedule automatically. For more detailed information of this Conference Navigator, please refer to [1, 26]. In conferences, talks are usually to present conferences papers. Hence, hereafter, talks and conference papers will be used interchangeably.

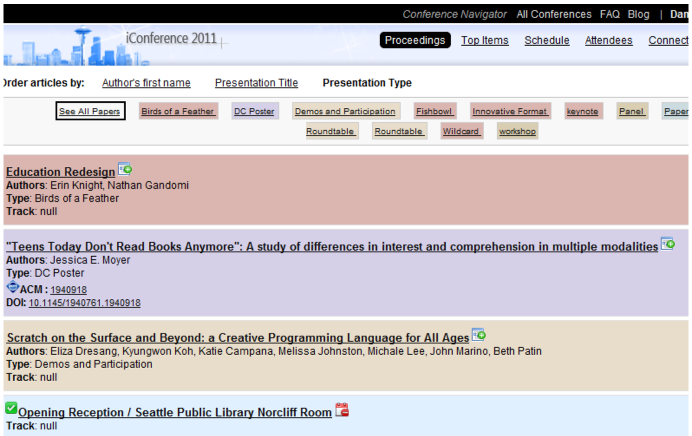

Figure 1. Main Page of Conference Navigator

II. RELATED WORK

There are two prominent technologies in recommendation field – content-based recommendations and collaborative filtering (hereafter, CF) recommendations.

The *content-based recommendation* is based on the assumption that 'the information favored in the past will be favored in the future.' This approach builds a user model according to content properties of users' favorite items and tries to find the most similar items according to the user model. Therefore, this approach is able to suggest recommendations with relatively small set of user preferences. However, this approach cannot be used in domains where the content information is unavailable such as pictures, video and music or resources highly relying on personal tastes such as jokes. This technology can easily suggest obvious and boring recommendations, as well. For instance, in news recommendations, it is very likely to recommend a series of similar or duplicated news items that happened recently [18].

The *CF recommendations* emerged as an attempt to automate the word-of-mouth in the age of Internet. The technology proved its worth in recommending taste-based items such as movies, jokes, music, etc., where the preference is hard to be appreciated by the contents. It became popular for its ability to suggest serendipitous and diverse recommendations. Due to the aforementioned strengths, the technology has been deployed in numerous systems in various domains [21]. Because this technology is based on information similarity from users to users, for the cold-start users who just start to use the system and have insufficient items to express their information preferences, it is hard to generate reasonable recommendations. Newly introduced items which don't have any user's ratings or which are bookmarked by very few users also suffer from the same problem. There is no chance to be a part of recommendations until the new items receive enough votes from users [2]. In addition, CF appeared to be not well-protected against malicious users who try to harm the system or to make a profit by gamming the system. For example, by copying the whole user profile, a malicious user is perceived by the system to be a perfect peer user and the products added by him are therefore recommended to the target user. Moreover, since recommenders have to compare all other users in order to find the peer group, the CF computation requires substantial off-line process [14].

As shown, both technologies have pros and cons. Accordingly, there are hybrid recommendation approaches fusing more than one technologies together. Melville, et al. (2002) and Adomavicius and Tuzhilin (2005) are exemplary studies to combine CF technologies with items' content features. Especially, Melvill, et al, (2002) partially inspired this paper.

As another way to solve the problems of typical CF, the recommendations based on users' social network are proposed. The researchers studying social network (SN)-based recommendations suggest that CF-related problems (e.g. cold-start problem, malicious users, high computational expenses) occur in part because the recommender systems make a choice of peer group purely by similarity computation, with no way for the target users to get involved in the recommendation process. The SN-based recommendation is to let users be a part of the recommendations by utilizing users' self-defined social connections. The target users will know who their peers are and this approach is known to be computationally more efficient than CF.

The main reason why the SN-based recommendations became possible nowadays is thriving online social networking sites. With the success of online networking sites, the kinds of social ties possibly used for SN-based recommendations variously ranged from undirected networks such as friendships [6], colleagues [12] and co-membership of a group [9] to directed network such as trust networks [22], following/watching relations [8], and email senders and recipients [20]. Up to now, however, most of SN-based recommendations take advantage of limited kind of social networks; trust-based social networks and friendships.

Golbeck (2009) says that the traditional definition of 'trust' is related to security and reliability, but the broader definition of trust for nowadays is related to 'a matter of opinion and perspective'. She refers to the broader trust as 'social trust'. She suggests that information can be aggregated, sorted and filtered through social trust [4]. In her own study, Golbeck also showed that users prefer recommendations from trusted people to CF recommendations [5]. Massa and Avesani's study (2004) showed that a user's trust network can solve the ad-hoc user problem, improve recommendation prediction and attenuate the computational complexity. Using Epinions data set, a trust-based technology generated more precise recommendations than CF technology. In addition, for users with 4 ratings (i.e. cold start user), trust-based technology could make the recommendations for 66% of these users, while CF could make recommendation for only 14% of the users with a higher margin of error [15]. Other studies indicated that a trust network decreases the recommendation error and increases the accuracy as well [17, 25]. For users with a unique taste, their own trusted network could increase the satisfaction of recommendations, since they are able to know where the information came from [23].

Pera and Ng (2011) introduced a SN-based recommendation algorithm which is usable for book recommendation domain. Their recommendation is a hybrid approach fusing metadata properties of books with users' social context. First in order to choose candidate books to recommend, the tag-based content similarities between the candidate books and target users' favorite books were counted, rather than the similarities of contents derived directly from the books (such as from the titles, abstracts or the authors' names). In the subsequent process, they aggregated the ratings of the candidate items given by the target users' friends. They also computed how the friends' tastes are similar to the target users. In the experiment using LibraryThing, they contrasted the SN-based recommendation with the CF recommendations provided by Amazon and content-based recommendations provided by LibraryThing, as baseline. As the result, the quality of the hybrid recommendations combining metadata information and friend relations was better than other two baseline approaches in terms of precisions and ranks [19].

Liu and Lee (2010) introduced a recommendation purely using users' friendships based on user study with a Korean online networking site, Cyworld. They suggested items based on typical CF approach (based on the nearest neighbors' preferences), SN-based approach (based on friend's preferences), and hybrid approach (based on the combination of both the nearest neighbors and users' friends). Even though they tried to augment the affects of social network more than the peer users with different weights, the naïve combination of peer and social connections performed the best. In addition, social network-based recommendation performed the worst [11].

To my best knowledge, there is no study for recommending conference talks. For example, using bibliographic management systems, some researchers tried to recommend scientific papers using the metadata of articles, such as authors' names, title, abstracts and keywords, and users' own social tags [3, 10, 13, 24]. However, these previous studies were not about conference talks, and the foci were papers, per se. This study is the first attempt to explore the conference talk recommendations.

III. Recommendation Description

As mentioned, recommending conference talks is a special domain. Even though the final outputs of this recommendation are simply talks, the talks embed implicit metadata such as authorship, abstract, title, etc. These items can be extended to social context, as well. They are core objects which users' social interactions in a conference are usually initiated with and centered on. Put differently, conference attendees usually start their conversations with other attendees regarding the papers they presented at the conference and share the related research interests through the papers. Therefore, we think that there are various aspects we need to consider in recommendations.

We considered three aspects – users' preferences on information items, social network of users and content of items – and generated recommendations using one of these aspects or hybrid approaches fusing all of them.

A. Basic Recommendation Approaches

The first two recommendations take advantage of preferences of conference attendees and their social context information. CF recommendations are based on all preferences of the whole population of our dataset, and SN-based recommendations are based on both user preferences and users' own social connections (i.e. their research collaborators).

The Conference Navigator system which our study is based on doesn't provide numeric rating mechanism when users bookmark conference talks. However, the system enables the users to express their interests on certain talks by bookmarking them. Hence, we encoded users' preferences of talks as binary ratings; 0 (i.e. no interest) or 1 (i.e. interest). For CF recommendations, the Jaccard similarity was used to compute bookmark similarities among users [9]. Based on this Jaccard similarity, we picked the most likely-minded users who have the highest bookmark similarity with our target users. We called them as 'peers'. Specifically, we limited the number of peers as top 5. That is to say, we take into account the preferences of five the most similar users. After computing the similarities, we chose candidate items in peer users' bookmarks, which are not bookmarked by our target user. Then in order to select the most presumably favorable items, we aggregated the similarities of the peers to whom each candidate item belongs. For instance, target user *A* has two peers – user *B* and user *C*. The information similarity of user *A* with user *B* and user *C* is 0.77 and 0.24, respectively. From user *B* and *C*'s bookmarks, we found item #2 and #3 are not bookmarked by *A*. The item #2 is bookmarked by both user *B* and *C* and #3 is bookmarked by only user *C*. Then in order to choose which one is more favorable to user *A*, we aggregated and averaged out the similarities of users who have the candidate items. Hence, the recommendation probability of #2 is 0.505 and the score of #3 is 0.24 and, conclusively, the former one is more recommendable.

$$\mathrm{CF}_{u,i} = \frac{\sum_{v \in p_u} Jaccard_{u,v}}{\mathrm{V}} \qquad (1)$$

Equation 1 indicates the CF recommendation probability of items *i* for a target users *u* by summing up the Jaccard similarity of *u*'s peer *v* who has the candidate item *i* in his bookmark. The variable *V* is the total number of peers who has the item *i*.

For SN-based recommendation, we built users' social network through their publication history. Using Scopus (http://www.scopus.com) system, we collected individual user's publication records. When two users have ever written a paper together, we assume that they are socially associated. Conference talks are highly related to their research interests; hence we suggest that users' co-authorship network is the best possible social network for our talk recommendations. In SN-based recommendations, we used the Jaccard similarity, as well. How to generate the recommendation is the same with CF recommendation, but only difference is that we substitute users' social connections for users' peers.

B. Content-boosted Recommendation Approaches

The next two recommendations are hybrid recommend-ations fusing users' bookmark preferences with contents of resources or with their social context. We used content-boosted collaborative filtering (CBCF) which is inspired by Melville, et al [16]. CBCF was designed for sparse datasets where simple co-rating profiles can't produce sufficient number of peers. Briefly, the original algorithm links one vector which consists of actual user ratings with another vector which consists of predicted ratings generated by keywords of the resources. Therefore, even when two users didn't rate exactly the same items, if they rated the similar items in the similar manner, these two users could be peer users to each other. While the Melvill's study was based on users' numeric ratings, our dataset has users' binary ratings of items. Hence, we introduced a modified CBCF algorithm.

In the selection of peers, our basic assumption is the exactly same with the Melville's study: to find peers who bookmarked the similar items. However, the difference with the previous study is that we didn't consider the degree of users' preferences because our study is solely based on bookmark history represented by binary ratings. Put differently, all items in a user's bookmark are equally important. Therefore, in comparing bookmark similarity between two users (i.e. our target users and their peers), we focused on how much two given users are interested in similar contents. Hence, the first stage is to compute the content-based profiles of all items in consideration. The content information of each paper is made up of multiple metadata, such as title, abstract, and keywords. Whole terms in these kinds of metadata were aggregated in one bag of words. Then, we pre-processed the content information through stemmer so as to reduce word variations to its stems or roots for the effective comparison. We chose Krovetz stemmer [7]. As the next step, we built an item vector having all terms and the TF-IDF values (term frequency – inverse document frequency) as a profile of each item.

We compared how two users' bookmark collections are similar to each other in item-by-item basis. When the same item was found in both bookmark collections, the match was counted as 1. For a target user's bookmarked item missing in his peer's bookmarks, among the peer's items, we picked the most similar one on the center of the item profiles. In particular, we used the cosine similarity to compute the closeness of two items. Then, we summed up content similarities between two users' bookmarks and averaged out the sum with the number of the target user's bookmarks (refer to equation 2) like the following content-boosted similarity computation (a.k.a., CBSim).

$$CBSim_{u,v} = \frac{\sum_{i=1}^{n} content\_sim_{ij}}{n} \qquad (2)$$

u and v denote our target user and his peer, respectively. The target user u has totally n bookmarked items and item i is one of them. j is one of the user v's bookmarked items which is the exact same item or the most similar with i. For instance, our target user *A* bookmarked talk #1 and #2 and his peer *B* bookmarked talk #1, #3 and #4. Because the talk #1 was bookmarked by both users, we counted the content similarity of that item as the value of 1. However, the talk #2 was bookmarked by the *user A* but not by the *user B*. We found that #3 is quite similar to #2, with the value of 0.9, and #4 is moderately similar, with the value of 0.5. Then we ignore the content similarity with #4 and, instead, only consider the talk #3. The resultant content-boosted similarity between users *A* and *B* ($CBSim_{A,B}$) is (1+0.9)/2.

We also altered the selection process of recommendation items from the Melville's original study. We took into account not only bookmark similarity between a target user and his peer users represented by content-boosted similarity (i.e. user-to-user), but also content similarity between target users' favorite items and candidate items (i.e. item-to-item). That is to say, we tried to choose items which the most likely-minded users favored, and are the most relevant to the target users' past favorites at the same time. Candidate items were selected among peers' bookmarked items which were missed in target user's bookmarks. Then, through item profiles, we computed the content-based similarity between a target user's bookmarked items and the candidate items. As the final recommendation probability, the content similarity was multiplied with average similarities of peers who bookmarked the candidate items. Equation 3 shows the equation.

$$\mathrm{CBCF}_{u,k} = \frac{CBSim_{u,v}}{V} \times \mathrm{Max}(ContentSimilarity_{ik}) \qquad (3)$$

where $CBCF_{u,k}$ denotes the CBCF recommendation probability of candidate item k for our target user u using the CBCF approach. First, for a candidate item k taken from the peer v's bookmark collection, we compared its content with u's all bookmarked items and found one item which holds the highest content similarity with the candidate item k. Then, we selected all V peers who have the candidate item k and averaged out their similarities with our target user u. This maximum content similarity and average similarities of peers were multiplied as the final probability.

In our content-boosted social network-based algorithm (CBSN), we generated the recommendations in the same way with the CBCF but substitute users' social networks for their likely-minded peer users.

Social Features Weights

In the CBSN, we also take into account users' social features indicating social strength – interaction frequency and structural equivalence – as an additional variation. We defined the recommendations as 'content-boosted social network recommendations with social features (CBSNS)'. For interaction frequency between social connections, we counted the frequency of co-authored papers. As mentioned, social network in our consideration was inferred from users' own publication history. If a pair of users has written a lot of papers together, we can assume that they are socially active and their interests may be overlapped to a large degree. For the equivalence of the social structure, we computed the number of shared connections between two users. When two users' social structures share many co-neighbors, it indicates that the connection of this pair is stronger than another pair which doesn't share any co-neighbors. We applied these two social features using the power weighting function [27] like equation 4.

$$\theta\left(\mathrm{CBSimilarity}_{u,v}\right) = \mathrm{Similarity}^{(f+e)} \qquad (4)$$

The variable *f* is the interaction frequency and *e* is the social equivalence. For a pair of our target user *u* and his social connection *v*, we modified the user-to-user similarity (equation 2) according to the social features. Then, we applied the same subsequent computation like the equation 3. Generally, through the power weighting function [27], the larger the sum of *f* and *e* is, the stronger the similarity is and vice versa.

C. *Community Vote-based Recommendation*

The last recommendation approach takes into account community's vote. Conferences focus on one research topic, area or discipline. Hence, conference attendees tend to form a community around the topic or the research area. If many attendees bookmarked an item, it is likely that the item has reasonably good quality appreciated by them. Therefore, rather than considering individual user's personal preferences, we recommended top 10, 5 and 2 popular items per conference. The reason why we selected these three top N numbers is that since the community vote-based recommendations are listed in the same manner with other recommendations, we tried to synchronize the results with other recommendation algorithms. Specifically, we counted whether the excluded test items of each target user are in a given top N item set or not. Finally, Figure 2 summarizes the design of our recommendations.

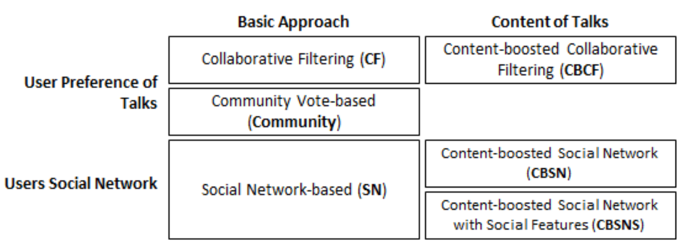

Figure 2. Recommendation Design

IV. EXPERIMENTAL EVALUATION

A. *Data Source*

For the evaluation of our algorithms, we used the bookmarking history of Conference Navigator. In this paper, among 16 conferences where the Conference Navigator was used, we considered bookmarking history of two conferences – ASIS&T 2010 Annual Meeting and iConference 2011, because the attendees were quite active in using our Conference Navigator. For these two conferences, we chose 126 target users who have at least two bookmarks. These users in our consideration bookmarked 11.4 talks on average. Among them, 33 users have at least one social connection with other users. As explained, we took advantage of users' co-authorship as their social networks. Their paper-authorship information was collected from scopus[1]. Even though the number users who have at least one own publication was 92, we found the social connections of only one of the third. The users having social connections have 23.1 their own papers and 1.7 co-authorship with 4.5 value of frequency on average. Put differently, each user wrote a paper with 1.7 other authors for 4.5 times. Table 1 is the descriptive statistics of the dataset.

TABLE 1. DESCRIPTIVE STATISTICS OF DATASET

No. of Distinct Conference Talks	296
No. of Users	126
No. of Users who have at least one own publication	92
No. of Users who have social connection(s)	33
No. of Bookmarks	1,456

B. *The Formal Evaluation*

In order to assess our recommendations, we used 10 cross-validation strategy. Among the whole collection of each user's bookmarks, we randomly split the collection into 10 equal-sized sets. This strategy excludes one set per iteration as a test set and generates recommendations. Then, we checked whether the suggested recommendations include the excluded items or not. If we find the test items, we count them as hits and otherwise, we count zero. The iteration continues for 10 times. In order to compute the accuracy of the recommendations, we count the number of hits according to three different ranks – the top 10, 5 and 2 recommendations. Recommendations are usually displayed in a ranked list. Users expected that items in a higher rank would be more important and accurate than other items in lower ranks. Therefore, it is critical to place correct predictions in a higher rank and the position of correct predictions is a critical evaluation criterion of recommendations. As the evaluation criteria, we calculated precision and recall based on the hit rates. Precision at point N (precision@N) is the ratio of the number of correctly predicted items in the top N list to N (refer to equation 5). Recall at point N (recall@N) is the ratio of the number of correctly predicted items in the top-N recommendation list to the total number of relevant items (equation 6).

$$\text{precision@N} = \frac{\text{No.of Correct Prediction}}{\text{Top N Set}} = \frac{\text{Test} \cap \text{Top N}}{\text{N}} \quad (5)$$

$$\text{recall@N} = \frac{\text{No. of Correct Prediction}}{\text{Size of Test Set}} = \frac{\text{Test } \cap \text{Top N}}{\text{Test}} \quad (6)$$

V. RESULTS

As the first results, Figure 3 and Figure 4 show the average precision and recall of all approaches except recommendations involving social networks. For all 126 target users, these three recommendations generated at least one recommendation.

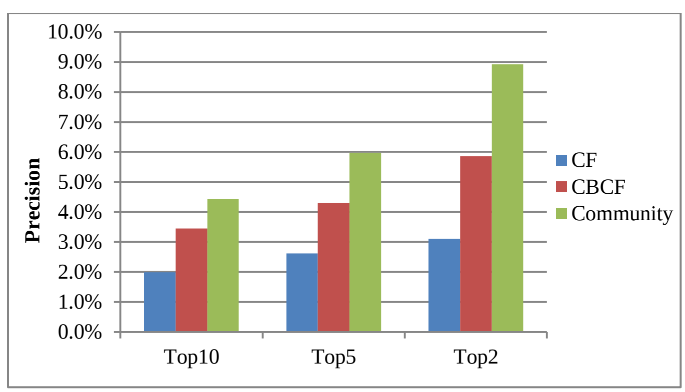

Figure 3. Precision Results of Top N, except Social Network-based Recommendations (for 126 users)

---

[1] http://www.scopus.com

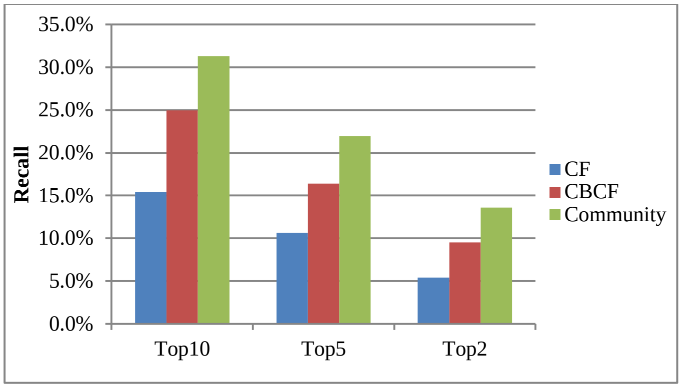

Figure 4. Recall Results of Top N, except Social Network-based Recommendations (for 126 users)

Regardless of the ranks (i.e. for all top 10, top 5 and top 2 recommendations), according to one-way ANOVA test, the precisions of non-personalized community vote-based recommendation were significantly higher than other two approaches (F = 21.5, p < .001 for the top 10; F = 14.5, p < .001 for the top 5; F = 11.1, p < .001 for the top 2). Content-boosted approach helped increase the precisions of the original CF recommendations, but the precision values were lower than the non-personalized community vote. The recall of all ranks yielded the same results (F = 13.2, p < .001 for the top 10; F = 7.9, p < .001 for the top 5; F = 6.0, p < .001 for the top 2). Even though we applied one of the most popular recommendation algorithms (i.e. CF), it couldn't beat the opinions of the majority of the research communities.

As explained in the section 4, in our dataset, there were only 33 users who have their collaboration relations in our dataset. We generated SN-based recommendations only for them. For other 93 users who don't have any social information, we couldn't generate any social network-related recommendations. Therefore, we compare the results of these 33 users separately including the social network-based recommendations. The Figure 5 and Figure 6 are the results.

The results of precisions show that, in lower ranks (i.e. top 10 and top 5), content-boosted SN-based recommendations (CBSN) could not beat the community vote-based recommendations, although the results were relatively good. However, for top 2 results where the most accurate recommendations should be placed, the CBSN recommendations preformed the best. Moreover, in most of the cases, the recommendations based on collaboration relations performed better than CF. The results were all significantly different (F = 1.82, p < .001 for the top 10; F = 1.57, p < .001 for the top 5; F = 0.98, p = .035 for the top 2). The results of the recalls show the exactly same pattern (F = 1.30, p < .001 for the top 10; F = 1.06, p < .001 for the top 5; F = 0.52, p < .001 for the top 2). That is to say, in terms of both accuracy and completeness of the recommendations, CBSN is the best recommendation approach to suggest conference talks.

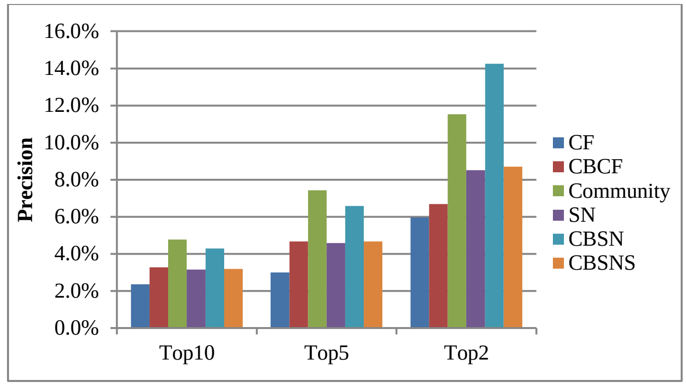

Figure 5. Precision Results of Top N including Social Network-based Recommendations (for 33 users)

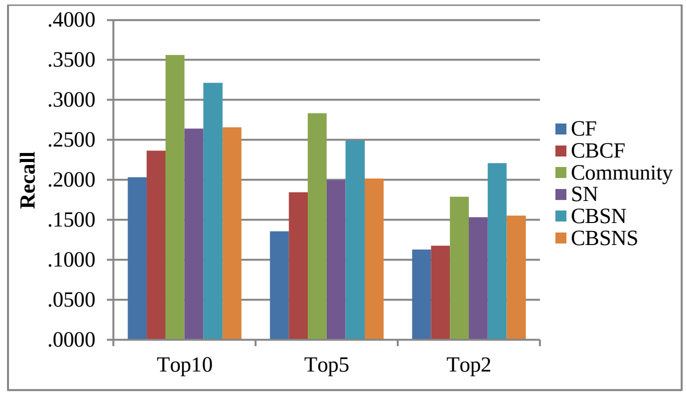

Figure 6. Recall Results of Top N including Social Network-based Recommendations (for 33 users)

In addition, boosting recommendations with content information looks beneficial to make the quality better. Content-boosted algorithms such as CBCF and CBSN have the higher precisions and recalls than the original algorithms which didn't have any aid of content features. However, adding social features in recommendations doesn't help improve the recommendation quality. It seems that social connections of users are good foundations for personalized recommendations, but the social context information per se is not important. In order to explore this idea more, we computed whether having many social partners as recommendation sources is helpful to make better recommendations. We computed the correlations of the number of social connections with the precision of the top 2 recommendations or with the recall of the top 2 recommendations. Unfortunately, we couldn't find any significant correlations (r = -0.08, p = 0.26 for top 2 precision; r = -0.12, p = 0.10 for top 2 recall). We interpreted this result to mean that regardless how many social connections a user has, the information of the social connections is valuable and enhance recommendation quality.

In spite of the good performance and popularity, one well-known shortcoming of CF is cold start user problem. Until users rated or bookmarked sufficient number of items [2], it is hard for them to get any CF recommendation or to receive reasonable quality of recommendations. A good alternative of CF recommendations for this cold-start user problem is to utilize content information of their favorite items. Even for a user who rated and bookmarked only one item, by referring to the content metadata, the recommendation systems are able to infer what the user's interest is and furthermore, to make suggestions for him. As shown, information items in our consideration are talks consisting of various textual information; hence we tested whether content-boosted recommendation approaches are really the most effective for cold-start users.

We compared the precisions and recalls of each recommendation depending on how many items each user bookmarked. First, we divided users into three groups; cold-start users ($n \leq 4$), users having moderate number of items ($n \leq 15$), and users having relatively larger number of items ($n > 15$). There were 26, 57, and 43 users in each group. Among the users who have social connections, 11, 10 and 12 users belong to each group, respectively. For the test of statistical significance of the results, we used one way ANOVA test ($p < 0.05$). Figure 7 shows the differences of recommendation precisions according to the user groups, and Figure 8 is about the recalls. First, we examined whether there are differences of precisions within the results of each group. For cold-start users, all kinds of the social network-based recommendations generated significantly better suggestions than other CF and community-based recommendations, across all rank levels of the precisions ($F = 1.66$, $p < .001$ for the top 10; $F = 1.41$, $p < .001$ for the top 5; $F = 1.29$, $p < .001$ for the top 2). Particularly, in higher rank, the CBSN approach produced the most accurate predictions for the cold-start users. For other two groups of users who have medium or large degree of items, the CBSN and community vote-based recommendations worked comparably well. Especially in higher rank results, the CBSN recommendations were the best approach.

However, for cold-start users who have insufficient bookmarks, the community votes, which are made up of collective intelligence of the conference attendees and produced generally good results for other two groups of users, were not useful. Instead, cold-start users tend to follow their social partners' opinions when looking for interesting papers. For other two groups of users, nonetheless, community vote-based recommendations produced the best or second best suggestions depending on the ranks. Therefore, for richer users who have sufficient number of bookmarks, the opinions of the majority are important source of information. In order to support this suggestion, we also ran correlations between the users' number of bookmarks and the quality of community vote-based recommendation. For all rank lists, there were significantly positive correlations ($r = .17$, $p < .001$ for Top 10, $r = .13$, $p < .001$ for Top 5, and $r = .06$, $p < .001$ for Top 2). That is to say, the more bookmarks a user has, the more helpful the community-based recommendations were. The results were also same in terms of recall.

VI. Conclusion and Discussion

In this paper, we examined various approaches to recommend talks at research conferences. The approaches we explored utilize three sources of information – user preferences about information items, information contents, and users' social connection. To our best knowledge, this is the first study related to the problem of recommending conference talks. In the 10 cross-validation test, we found that content-boosted social network-based approach and community vote-based approach performed the best. In higher rank results, the content-boosted social network-based recommendations especially outperformed other approaches. In addition, for the cold-start users who have insufficient number of bookmarks to receive reasonable recommendations, the content-boosted social network-based recommendations were always the most effective way to personalize the information across all ranks.

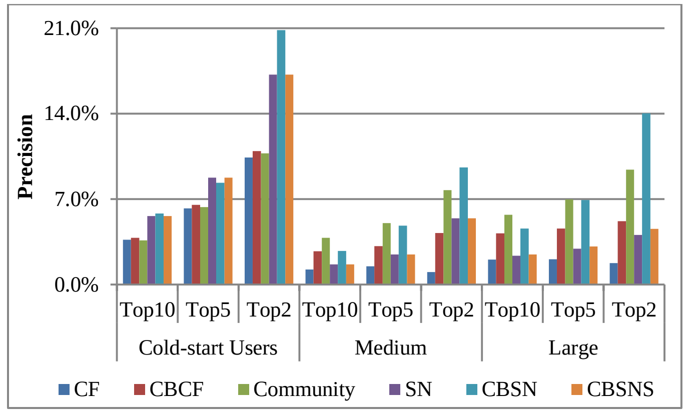

Figure 7. Precisions according to User Group by the Number of Their Bookmarks

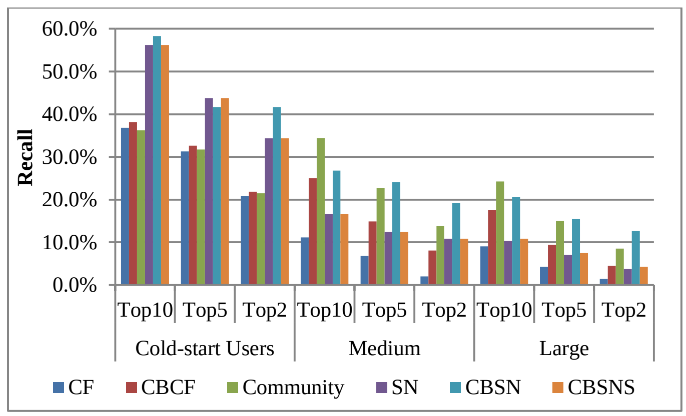

Figure 8. Recalls according to User Group by the Number of Their Bookmarks

In this study we used co-authorship connections to form users' social network. When two users collaborated and wrote papers together, it is very likely that their research interests and topics are largely overlapped. Therefore, the favorite items of users' colleagues are effective sources to acquire interesting information. Moreover, we interpreted the good performance of the community vote-based approach to mean that majority of conference attendees understand conference talks/papers in the similar manner as a research community. Because they usually have background knowledge of the conference topics or areas, conference attendees have some insights to discover interesting papers, and their insights are helpful to other attendees, as well. In addition, they are the users who constructed the collective intelligence. The increasing performance of community vote-based recommendations along with the increasing number of bookmarks supports our suggestions. On the other hand, the community vote-based approach wasn't helpful for the cold-start users. Cold-start users don't have enough bookmarks to follow and form the votes of the majority.

The study is based on adaptive conference navigation supporting system – Conference Navigator and exploited data from two conferences that actively used the Conference Navigator system. The system, however, has data about other conferences, as well. Therefore, in future, by examining the whole users' usage logs, we might be able to trace whether there are any preference changes before, during and after conferences. In addition, we will expand the item of recommendation from talks to people to interact with through a conference. We also plan to use more sophisticated way to fuse various aspects of conference talks and how to improve the recommendation quality. Lastly, we found that SN-based approaches have some limitation to generate recommendations. The very basic assumption of the SN-based recommendation is that users have to have their own social network in the system or the system should be able to infer their social connections in somehow. In this study, even though we picked about 130 target users, we were able to suggest SN-based recommendations only for one of the fourths. We will expand users' social network outside of our system such as online social networking applications like LinkedIn and Facebook.

References

[1] Brusilovsky, P., D. Parra, S. Sahebi, and C. Wongchokprasitti, Collaborative information finding in smaller communities: The case of research talks. Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), 2010 6th International Conference on, 2010: p. 1-10.

[2] Desrosiers, C. and G. Karypis, A Comprehensive Survey of Neighborhood-based Recommendation Methods, in Recommender Systems Handbook, R. Francesco, et al., Editors. 2011, Springer. p. 107-144.

[3] Firan, C.S., W. Nejdl, and R. Paiu, The Benefit of Using Tag-Based Profiles, in Proceedings of the 2007 Latin American Web Conference. 2007, IEEE Computer Society. p. 32-41.

[4] Golbeck, J., Introduction to Computing with Social Trust, in Computing with Social Trust, J. Golbeck, Editor. 2009, Springer: London. p. pp. 1 - 5.

[5] Golbeck, J. and J. Hendler, Inferring binary trust relationships in Web-based social networks. ACM Trans. Internet Technol., 2006. 6(4): p. 497-529.

[6] Groh, G. and C. Ehmig, Recommendations in taste related domains: collaborative filtering vs. social filtering, in Proceedings of the 2007 international ACM conference on Supporting group work. 2007, ACM: Sanibel Island, Florida, USA. p. 127-136.

[7] Krovetz, R., Viewing morphology as an inference process, in Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval. 1993, ACM: Pittsburgh, Pennsylvania, United States. p. 191-202.

[8] Lee, D. and P. Brusilovsky. Improving Recommendations Using Watching Networks in a Social Tagging System. in The Proceedings of IConference 2011. 2011. Seattle, WA, USA.

[9] Lee, D.H. and P. Brusilovsky, Using self-defined group activities for improvingrecommendations in collaborative tagging systems, in Proceedings of the fourth ACM conference on Recommender systems. 2010, ACM: Barcelona, Spain. p. 221-224.

[10] Liang, H., Y. Xu, Y. Li, and R. Nayak, Tag Based Collaborative Filtering for Recommender Systems, in Proceedings of the 4th International Conference on Rough Sets and Knowledge Technology. 2009, Springer-Verlag: Gold Coast, Australia. p. 666-673.

[11] Liu, F. and H.J. Lee, Use of social network information to enhance collaborative filtering performance. Expert Systems with Applications, 2010. 37(7): p. 4772-4778.

[12] Maltz, D. and K. Ehrlich, Pointing the way: active collaborative filtering, in Proceedings of the SIGCHI conference on Human factors in computing systems. 1995, ACM Press/Addison-Wesley Publishing Co.: Denver, Colorado, United States. p. 202-209.

[13] Marlow, C., M. Naaman, D. Boyd, and M. Davis, HT06, tagging paper, taxonomy, Flickr, academic article, to read, in Proceedings of the seventeenth conference on Hypertext and hypermedia. 2006, ACM: Odense, Denmark. p. 31-40.

[14] Massa, P. and P. Avesani, Trust-Aware Collaborative Filtering for Recommender Systems. Vol. 3290. 2004. 492-508.

[15] Massa, P. and P. Avesani, Trust-aware recommender systems, in Proceedings of the 2007 ACM conference on Recommender systems. 2007, ACM: Minneapolis, MN, USA. p. 17-24.

[16] Melville, P., R.J. Mooney, and R. Nagarajan, Content-boosted collaborative filtering for improved recommendations, in Eighteenth national conference on Artificial intelligence. 2002, American Association for Artificial Intelligence: Edmonton, Alberta, Canada. p. 187-192.

[17] O'Donovan, J. and B. Smyth, Trust in recommender systems, in Proceedings of the 10th international conference on Intelligent user interfaces. 2005, ACM: San Diego, California, USA. p. 167-174.

[18] Pazzani, M. and D. Billsus, Content-Based Recommendation Systems, in The Adaptive Web: Methods and Strateiges of Web Personalization, P. Brusilovsky, A. Kobsa, and W. Nejdl, Editors. 2007, Springer: Berlin, Germany. p. 325-341.

[19] Pera, M.S. and Y.-K. Ng, With a Little Help from My Friends: Generating Personalized Book Recommendations Using Data Extracted from a Social Website, in Proceedings of the 2011 IEEE/WIC/ACM Joint Conference on Web Intelligent (WI'11). 2011: Lyon, France. p. pp. 96-99.

[20] Roth, M., A. Ben-David, D. Deutscher, G. Flysher, I. Horn, A. Leichtberg, N. Leiser, Y. Matias, and R. Merom, Suggesting friends using the implicit social graph, in Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining. 2010, ACM: Washington, DC, USA. p. 233-242.

[21] Schaefer, J.B., D. Frankowski, J. Herlocker, and S. Sen, Collaborative Filtering Recommender Systems, in The Adaptive Web: Methods and Strateiges of Web Personalization, P. Brusilovsky, A. Kobsa, and W. Nejdl, Editors. 2007, Springer: Berlin, Germany. p. 291-324.

[22] Shi, Y., M. Larson, and A. Hanjalic. Towards Understanding the Challenges Facing Effective Trust-Aware Recommendation. in Proceedings of the 2nd ACM RecSys Workshop on Recommender Systems and the Social Web. 2010. Barcelona, Spain.

[23] Tintarev, N. and J. Masthoff, Effective explanations of recommendations: user-centered design, in Proceedings of the 2007 ACM conference on Recommender systems. 2007, ACM: Minneapolis, MN, USA. p. 153-156.

[24] Tso-Sutter, K.H.L., L.B. Marinho, and L. Schmidt-Thieme, Tag-aware recommender systems by fusion of collaborative filtering algorithms, in Proceedings of the 2008 ACM symposium on Applied computing. 2008, ACM: Fortaleza, Ceara, Brazil. p. 1995-1999.

[25] Walter, F.E., S. Battiston, and F. Schweitzer, Personalised and dynamic trust in social networks, in RecSys, D.B. Lawrence, et al., Editors. 2009, ACM. p. 197-204.

[26] Wongchokprasitti, C., P. Brusilovsky, and D. Parra, Conference Navigator 2.0: Community-Based Recommendation for Academic Conferences, in Proceedings of Workshop on Social Recommender Systems at the International Conference on Intelligent User Interfaces (IUI 2010). 2010: Hong Kong, China.

[27] Xia, C., X. Jiang, S. Liu, Z. Luo, and Z. Yu, Dynamic item-based recommendation algorithm with time decay. Natural Computation (ICNC), 2010 Sixth International Conference on, 2010. 1: p. 242-247.