2010 10th IEEE International Conference on Computer and Information Technology (CIT 2010)

Impact of Peers' Similarity on Recommendations in P2P Systems

Loubna Mekouar
University of Waterloo
Waterloo, Canada
Email: lmekouar@bbcr.uwaterloo.ca

Youssef Iraqi
Khalifa University
Sharjah, UAE
Email: Youssef.Iraqi@kustar.ac.ae

Raouf Boutaba
University of Waterloo
Waterloo, Canada
Email: rboutaba@bbcr.uwaterloo.ca

***Abstract*—In this paper, we propose a novel recommender framework for P2P file sharing systems. The proposed recommender system is based on *user-based collaborative filtering* technique. We take advantage from the partial search process used in partially decentralized systems to explore the relationships between peers. The proposed recommender system does not require any additional effort from the users since implicit rating is used. To measure the similarity between peers, we investigate similarity metrics that were proposed in other fields and adapt them to file sharing P2P systems. We analyze the impact of each similarity metric on the accuracy of the recommendations. Files' recommendations will increase users' satisfaction since they will receive recommendations on files that they prefer.**

***Keywords*-Recommender systems; P2P systems; User-based collaborative filtering; Similarity metrics;**

I. INTRODUCTION

Recommender systems are widely used in e-Commerce applications (e.g., amazon.com, BizRate.com, Epinions.com, Yahoo.com) [1]. Recommender systems take advantage of the collected data that represents customers' experiences to predict their future needs. These systems suggest products and services that most likely will be of interest to the customers. The collaborative filtering recommender techniques are achieving widespread success on the web [1], [2].

In P2P file sharing systems, users are overwhelmed by a large collection of files available for download. Unfortunately, finding files of interest is time consuming. Recommender systems suggest to users files based on their profile. These users will be motivated to download the recommended files and hence, will remain active members. While they are downloading the files, they will upload files to others increasing their contribution to the system.

We propose a novel recommender framework that will help peers find relevant files of interest based on peers' profile. The profile reflects their past choices, experiences and preferences. The proposed recommender system is based on *collaborative filtering*. Peers collaborate to filter out irrelevant files and find interesting ones.

The paper is organized as follows. Section II presents the motivation behind this work. Section III presents the proposed recommender framework. Section IV describes the similarity metrics adopted in this work. Section V describes the performance evaluation conducted and presents the results. Section VI presents an analysis of different similarity metrics and their impact on the recommendations' accuracy. Finally, Section VII concludes the paper.

II. MOTIVATION

The main known problems of collaborative filtering are the followings [1], [2]:

- *Cold start*: This problem occurs for a new user or at the start of the system. It is difficult to make recommendations for a new user based on users' similarities since no rating is provided yet or the user's profile is not known yet.
- *Popularity effect*: This problem occurs when the given recommendations are obvious and evident from the user's point of view.
- *Data sparseness*: This problem occurs when only few users have rated few items. It is difficult to predict the user's interests and make accurate recommendations.
- *Trust*: This problem occurs when untrustworthy users provide false ratings. The system should be able to choose only highly reputable users while making recommendations. This will reduce the impact of untrustworthy users that influence badly the recommendation accuracy and hence, will increase the trust given by the peers to the recommender system.

Although, e-Commerce applications have been using recommender systems for at least a decade, this research field is still a fertile area in P2P systems. Only few research works have addressed recommender schemes in P2P systems [3], [4], [5].

In [3], the authors propose a decentralized recommendation system that takes advantage of the high clustering coefficient of Preference Networks. In [4], the authors propose a distributed collaborative filtering method that is self-organizing and operates in a distributed way. These recommender schemes are suitable for decentralized P2P systems but not for partially decentralized systems. In addition, theses schemes generate a significant amount of overhead to make files' recommendations. As an example, in [5], it is required to maintain the following lists by each peer in the system: the top-N most similar users, the top-N most fresh random IP addresses and the K most recently visited users.

978-0-7695-4108-2/10 $26.00 © 2010 IEEE
DOI 10.1109/CIT.2010.94

The periodic exchange and update of information between peers is costly.

In this paper, we propose to explore relationships between peers by taking advantage from the partial search process used in partially decentralized systems. In order to make personalized recommendations, the implicit rating approach is used and hence, no additional effort is required from the users.

Similarity has been used in many fields like natural and social sciences as well as engineering and statistics. Several metrics have been proposed to compute similarity. In this paper, we investigate several similarity metrics in the context of P2P recommender systems. We adapt these metrics to the context of file sharing P2P systems. We investigate these similarity metrics in both the weighted and non weighted techniques. The impact of each similarity metric on the accuracy of the recommendations is analyzed.

III. The proposed Recommender Framework

In e-Commerce applications, the collaborative filtering technique is based on the ratings of the products provided by the customers. In P2P file sharing systems, the collaborative filtering technique can be used based on the ratings of the files provided by the users.

A. Implicit Rating versus Explicit Rating

After downloading a file, two rating approaches can be considered: explicit rating and implicit rating. In the *explicit rating* approach, the user has to explicitly provide a rating for each file she/he downloads according to its content (i.e., matches the user's preferences or not). This approach necessitates an additional effort from the users. A rating scheme from 1 (not interesting at all) to 5 (very interesting) can be useful to assure recommendation accuracy. Users have to provide their ratings for different files to enrich the system with different opinions and experiences. Since *explicit rating* solicits an additional effort from users, it is difficult to enforce, especially in systems where 70% of peers are free riders [6]. This approach will likely suffer from the *Cold start* and *Data sparseness*. Also, *explicit rating* provides malicious peers with a way to influence the rating system which may lead to the *Trust* issue described in section II.

The *implicit rating* approach does not require the users to rate the files. It assigns ratings implicitly. The fact that ratings are generated automatically without involving users, alleviate them from the burden of explicitly providing ratings for each file they have downloaded. We propose to assign a rating of 1 (*I like it*) to the files owned by the user. All other files are assigned a rating of 0 (*I do not know*). Note that a rating of 0 does not mean that the user does not like the file.

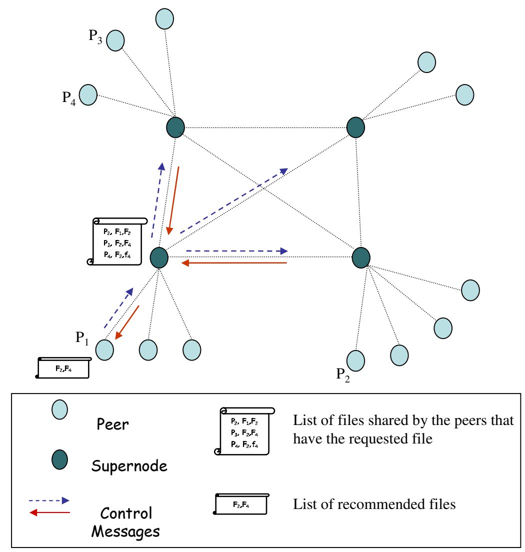

Figure 1. The Proposed Recommender Framework

B. User-based versus Item-based Collaborative Filtering

Figure 1 depicts an example of the information flow between the peer P_1 requesting a file and its supernode. After receiving a request from peer P_1, and assuming the file is not found locally, its supernode sends a request to other supernodes. These supernodes will send back the search result which is a list of peers that have the requested file and the files that these peers are sharing. Based on this information, the supernode of P_1 will use the proposed recommender scheme to generate a list of recommended files.

Since recommendations are given to peers in real time, it is preferable to explore relationships between peers rather than between files. We take advantage from the partial search process used in partially decentralized systems. The partial search performed by supernodes limits the number of peers in the search result. This number is much less than the number of files shared by all the peers in the system. In addition, finding relationships between all files is time consuming and is usually done offline. For these reasons, adopting user-based collaborative filtering in P2P systems is more practical than using item-based collaborative filtering algorithms.

Figure 2 depicts the steps required in the proposed framework to make recommendations to the peers. During the life cycle of a transaction in a P2P system, the following steps are performed:

1) Send a file request
2) Receive a list of peers that have the requested file
3) Use similarity metric to choose most similar peers to the peer requesting the file (the active peer)
4) Use the weighted or non weighted files' popularity to

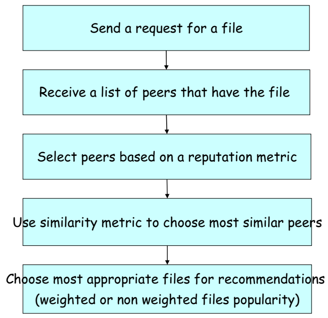

Figure 2. Recommendation Transaction Life Cycle

choose most appropriate files for recommendations. The non weighted file popularity approach selects the most popular files among the selected similar peers independently from how similar the peers are to the active peer. The weighted approach uses the similarity metric to compute a weighted file popularity before suggesting files for recommendation.

The similarity metrics considered in this work are used during step 3, and the weighted and non weighted approaches have been enforced in step 4.

IV. Similarity Metrics and Binary Ratings

Similarity has been used in data mining, pattern recognition, information retrieval, information theory, data clustering and artificial intelligence.

The most used similarity techniques for recommender systems are the Pearson correlation and the Cosine measure [2], [1]. However, a thorough investigation of similarity metrics based on binary ratings reveals the existence of a number of other potentially better similarity metrics.

Adopting an implicit rating approach, implicates a binary value (i.e., 1 if the peer has the file, 0 otherwise) and hence promotes the use of similarity measures for binary data.

Different similarity metrics have been used in exploratory data analysis [7], and in genetics and molecular biology [8].

We adopt the following notations:

Let p_i be the active peer (i.e., the peer requesting the file).

Let p_j be the peer for which we want to compute the similarity with the active peer p_i.

For a particular file f, let C be the observation that the active peer p_i has the file f. And let D be the observation that a peer p_j has this file.

Let a, b, c, and d as follows:

- a: number of times $C = 1$ and $D = 1$. This represents the number of files common to both p_i and p_j.
- b: number of times $C = 1$ and $D = 0$. This represents the number of files owned by p_i but not p_j.
- c: number of times $C = 0$ and $D = 1$. This represents the number of files owned by p_j but not p_i.
- d: number of times $C = 0$ and $D = 0$. This represents the number of files neither owned by p_i nor p_j.

The similarity metrics may be grouped into two classes according to how they deal with the negative co-occurrence (i.e., d value). These are the metrics that use the d value in their equation. Table I shows the similarity metrics that consider the negative co-occurrence, while table II shows the similarity metrics that do not consider this co-occurrence. In [9], the similarity metrics in the former table are named type 2 similarity metrics, while those in the latter table are named type 1 similarity metrics.

Each similarity metric has its own characteristics and properties. In this paper, we explore all these similarity metrics by applying them to find the most similar peers in order to make appropriate recommendations. We also investigate both the weighted approach and the non-weighted approach in computing the recommendations. We want to analyze the impact of the similarity metrics on the recommender system. Furthermore, we study these similarity metrics under different scenarios to evaluate their performance and their ability to make accurate recommendations.

It is important to note that since implicit rating is used for files' recommendations, the use of Pearson correlation is not applicable since the average rating given by a peer p to its files is always 1. In this case, the Pearson correlation measure is not well defined.

V. Performance Evaluation

A. Simulated Schemes

In this paper, we simulate the following techniques using the non weighted and weighted rating approaches:

- Scheme 3: *Asymmetric Peers' Similarity with File Popularity (ASFP)* [10]. This metric uses the following equation: $\frac{a}{a+b}$
- and the following schemes: *Ochiai I* (OcI), *Jaccard* (Jac), *Simple Matching* (SM), *Rogers and Tanimoto* (RT), *Ochiai II* (OcII), *Sokal and Sneath* (SS), *Anderberg* (And), *CzekanowskySorensen-Dice* (CSD), *Kulczynski II* (KII) and *Russel Rao* (RR) presented in tables I and II.

In the *Cosine measure* technique, the active peer and any other peer are represented by two vectors (generated from the list of files they own) and the similarity between them is measured by computing the cosine of the angle between the two vectors. In Binary rating, the *Cosine measure* and *Ochiai I* are equivalent. We simulate *Ochiai I*.

The goal from these simulations is to compare the performance of the presented schemes in terms of providing accurate files' recommendations.

B. Simulation Parameters

The simulation parameters are the following:

- We simulate a system with 1,000 peers and 1,000 files.

Similarity metric	Equation	Scheme number
Rogers and Tanimoto	$\frac{a+d}{a+d+2(b+c)}$	4
Simple Matching	$\frac{a+d}{a+b+c+d}$	5
Ochiai II	$\frac{ad}{\sqrt{(a+b)(a+c)(d+b)(d+c)}}$	6
Sokal and Sneath	$\frac{2(a+d)}{2(a+d)+b+c}$	7

Table I
SIMILARITY METRICS WITH NEGATIVE CO-OCCURRENCE

Similarity metric	Equation	Scheme number
Ochiai I	$\frac{a}{\sqrt{(a+b)(a+c)}}$	1
Jaccard	$\frac{a}{a+b+c}$	2
Anderberg	$\frac{a}{a+2(b+c)}$	8
CzekanowskySorensen-Dice	$\frac{2a}{2a+b+c}$	9
Kulczynski II	$\frac{a}{2}(\frac{1}{a+b}+\frac{1}{a+c})$	10

Table II
SIMILARITY METRICS WITHOUT NEGATIVE CO-OCCURRENCE

- At the beginning of the simulation, each peer has several files and each file has at least one owner.
- Peers are divided into four interest categories (C1: Action, C2: Romance, C3: Drama and, C4: Comedy) and files are also divided into the same four categories.
- The percentage of peers in each category is 25% and the percentage of files in each category is 25%.
- Each peer belongs to one category. Peers prefer to have most of the files from their category and only few files from other categories. We investigate the different schemes using different probabilities termed *Initial Profile* (0.5, 0.6, 0.7, 0.8, 0.9, and 1) leading to 6 scenarios. In the case of 0.9 for example, initially, each peer will have files from the category that she/he prefers with a probability of 0.9 and files from other categories with a probability of 0.1.
- If no file is recommended, file requests follow the real life distribution observed in [11].
- The threshold for each similarity metric is set to 0.1. This means that the similarity of a peer should be greater than 10% for the peer to be considered.
- We simulate 50,000 requests for each simulation.

Our simulations were implemented using the peer-to-peer simulator PeerSim [12]. The simulations were repeated several times for each scheme and for each *Initial Profile* probability. The results presented are the average values. Each scheme has been simulated using the weighted and non weighted rating techniques.

The performance metrics used in the literature are called *Recall* and *Precision*. While *precision* represents the probability that a recommended item is relevant, *recall* represents the probability that a relevant item will be recommended. In [13], recall is measured by taking into account the number of hits. A hit is considered when an item from the top N recommended files is in the test set. Usually, $N = 10$ is the number of items returned to users. The greater is N, the greater is the value of *Recall*.

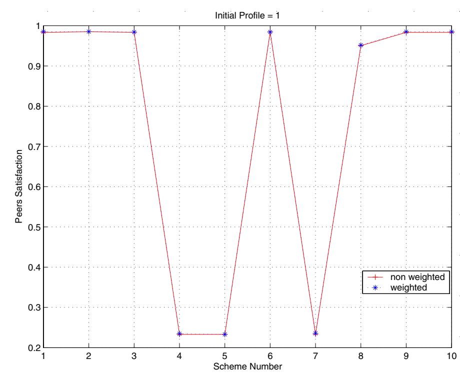

Figure 3. Peers Satisfaction (first scenario)

In these simulations, we limit the N value to only 1 item. This will make it hard to get a hit. For each scheme, we compute the *Peer Satisfaction*. This value is computed for all peers' categories and it represents the average value of the ratio between the number of recommended files that match peer's category over all the files recommended to the peer. Our goal is to assess accurately the effectiveness of the proposed recommender system and the used similarity metrics.

C. Simulation Results

We simulated all the schemes under the same conditions and we compared the performance of these schemes. The simulations were conducted in six different scenarios based on the *Initial Profile* probability. Because of space constraints, we present detailed results for three scenarios only.

1) First Scenario: At the beginning of the simulations, peers get files from the category that they prefer with a probability of 1 and no file from other categories is selected. Figure 3 depicts the peers' satisfaction for all the schemes with the non weighted and weighted rating approaches. By comparing the results obtained, there is no significant difference between these approaches. Peers' satisfaction almost reaches 100% for the following schemes: Ochiai I (1), Jaccard (2), ASFP (3), Ochiai II (6), CzekanowskySorensen-Dice (9) and Kulczynski II (10). For the Anderberg (8) scheme, peers' satisfaction is relatively lower (95%). However, this value decreases significantly for the following schemes: Rogers and Tanimoto (4), Simple Matching (5), Sokal and Sneath (7). This peers' satisfaction is settling around 23%. In these schemes, 98% of files that have been downloaded by the peers were recommended to them. The bad performance of these schemes can be explained by the fact that their corresponding similarity metrics take into account the negative co-occurrence as explained in table I. However, the fact that two peers do not have a specific file, does not mean that they do not like it. Also, it does not mean that they have the same interests.

2) Second Scenario: To show the effectiveness of the proposed schemes, we performed another set of simulations. In this scenario, peers start with files that match their category with an *Initial Profile* probability equals to 0.7. Figure 4 presents the results. Peers satisfaction is still higher for the following schemes: Ochiai I (1), ASFP (3), Ochiai II (6), CzekanowskySorensen-Dice (9) and Kulczynski II (10) compared to other schemes. Peers' satisfaction is decreased when using the Jaccard scheme (2) to achieve only 91% in the non weighted rating. A significant decrease in the performance of the Anderberg scheme (8) is also noticed in this scenario. The peers' satisfaction is only 82% in the non weighted rating approach which is slightly higher than the weighted rating approach. As mentioned in the previous scenarios, the following schemes: Rogers and Tanimoto (4), Simple Matching (5), Sokal and Sneath (7) do not provide good recommendations to the peers.

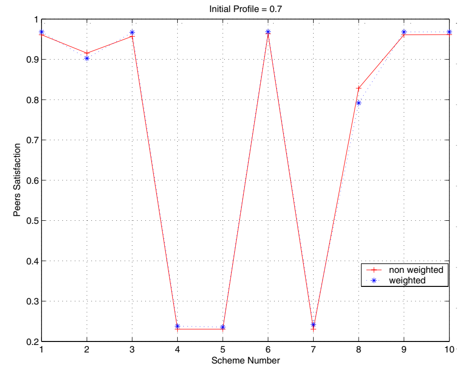

Figure 4. Peers Satisfaction (second scenario)

A decrease of the *Initial Profile* value to 0.7 will not lead to a significant decrease in *Peers Satisfaction* while using the weighted and non weighted rating approaches.

3) Third Scenario: Decreasing the value of *Initial Profile* to 0.6 allows to distinguish among the schemes that provide better recommendations to the peers. Figure 5 depicts the peers' satisfaction for all the schemes. The results are not as good as in the previous set of simulations. Peers' satisfaction is approximatively 70% by using the non weighted rating for the following schemes: Ochiai I (1), Jaccard (2), ASFP (3), CzekanowskySorensen-Dice (9) and Kulczynski II (10). However, despite of the low value of *Initial Profile* probability, the peers' satisfaction value is still acceptable. The recommender scheme Ochiai II (6) shows a significant increase in peers' satisfaction compared to the previously mentioned schemes. In the Ochiai II (6) scheme, peers satisfaction achieves a high score equals to 79%. The performance of this scheme in this scenario surpasses all other schemes. The Anderberg scheme (8) is less accurate in making recommendations. As discussed in the previous scenario, the following schemes: Rogers and Tanimoto (4), Simple Matching (5), Sokal and Sneath (7), are the worst schemes in making recommendations.

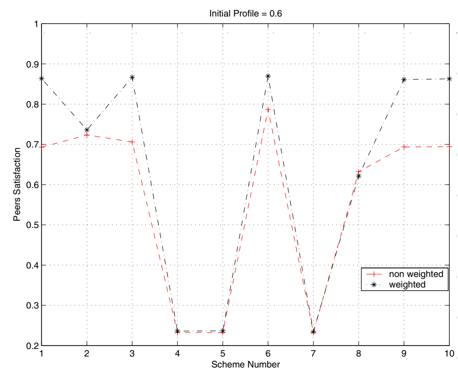

Figure 5. Peers Satisfaction (third scenario)

Figure 5 shows the good performance of the following recommender schemes using the weighted rating approach: Ochiai I (1), ASFP (3), CzekanowskySorensen-Dice (9) and Kulczynski II (10). These schemes surpass the other schemes in providing appropriate and accurate recommendations. Although the value of *Initial Profile* probability is relatively lower, the use of the weighted rating technique allows these schemes to make a good distinction between files' categories and recommend the appropriate files based on peers' profiles. Peers' satisfaction reaches 87% in contrast to 79% in the non weighted approaches. In general, the weighted rating techniques provide better recommendations' accuracy compared to the non weighted rating techniques.

Probability	Oc I	Oc I W	Jaccard	Jaccard W	ASFP	ASWFP	RT	RT W	SM	SM W
1	98.34	98.50	98.49	98.53	98.37	98.42	23.30	23.46	23.30	23.30
0.9	98.22	98.22	98.18	98.25	98.21	98.25	23.13	23.38	23.13	23.53
0.8	97.81	97.88	96.77	96.77	97.69	97.72	23.28	23.60	23.28	23.53
0.7	96.08	96.81	91.54	90.25	95.72	96.69	23.05	23.80	23.05	23.56
0.6	69.30	86.36	72.33	73.59	70.61	86.66	23.22	23.60	23.22	23.66
0.5	27.35	36.32	27.45	31.31	26.74	34.58	23.65	23.72	23.65	23.82

Table III
SUMMARY OF PEERS' SATISFACTION

Probability	Oc II	Oc II W	SS	SS W	And	And W	CSD	CSD W	K II	K II W
1	98.46	98.44	23.30	23.58	95.03	95.16	98.34	98.50	98.34	98.48
0.9	98.08	98.13	23.13	23.54	91.87	91.44	98.22	98.21	98.22	98.24
0.8	97.90	97.96	23.28	23.71	88.60	87.31	97.81	97.89	97.81	97.90
0.7	96.33	96.84	23.05	24.19	82.84	79.19	96.09	96.82	96.15	96.81
0.6	78.72	86.97	23.22	23.44	63.22	62.13	69.35	86.08	69.42	86.28
0.5	27.59	35.15	23.65	23.56	31.55	34.27	27.41	36.79	27.26	36.77

Table IV
SUMMARY OF PEERS' SATISFACTION (COND.)

VI. ANALYSIS OF RECOMMENDER SCHEMES

Similarity metrics aim at quantifying the extent to which objects resemble each other. Similarity metrics make possible to determine if the compared peers can be assigned to the same class or not.

Similarity metrics express the proportion of matches between two peers in a different way. While *Anderberg* and *Rogers and Tanimoto* similarity metrics give twice weight to disagreement, the similarity metrics *Sokal and Sneath* and *CzekanowskySorensen-Dice* give more weight to agreement. Also, the *Sokal and Sneath* metric is similar to the *Simple Matching* metric but gives double weight to matches. Similarly, *CzekanowskySorensen-Dice* metric is similar to the *Jaccard* similarity metric but gives twice the weight to matches. In table II, most of the similarity metrics are increasing functions of a and decreasing functions of b and c. Similarity is higher when the compared peers share more common files and have few distinctive files. While, similarity metrics in table I take into consideration the intersection, the differences and also the intersection of the complementary sets of the compared peers. For these metrics, the common files and the absence of same files have the same role. In addition to the common files, the absence of same files increases the similarity between the compared peers. However, the *Russel and Rao* metric is more severe in attesting the resemblance between peers, since the absence of same files is added only in the denominator. In this metric, similarity is based only on the common files owned by the compared peers over all the files.

In many applications such as image retrieval, the user is interested in the list of objects most similar to its request (ordered-based approach) rather than the values of the similarity scores (value-based approach) [9]. The similarity scores are not as important as the order of similar objects. However, in the performed simulations, we were interested to know the order of similar peers to the requester peer (i.e., to whom the recommendation is made) in addition to the similarity scores. The similarity of a peer should be greater than 10% to be considered. We considered both the ordered-based and the value-based comparisons in the simulations to assess the performance of the similarity metrics.

Tables III and IV present a summary of the results of the simulations. By comparing all the schemes using the weighted and non weighted rating techniques in terms of peers' satisfaction, we found that the following schemes: Ochiai I (1), ASFP (3), Ochiai II (6), CzekanowskySorensen-Dice (9) and Kulczynski II (10), provide better performance in terms of recommendations' accuracy. The Jaccard (2) and the Anderberg (8) schemes are less accurate. However, a low performance in providing appropriate recommendations is observed for the following schemes: Rogers and Tanimoto (4), Simple Matching (5), and Sokal and Sneath (7). From the tables III and IV, it is also clear that as the initial distribution of files becomes fuzzy, the schemes are not able to clearly find the exact peers' profile and hence will lead to poor peers' satisfaction. Moreover, the weighted rating technique improves the performance of the schemes since the weight of similarity measures is taken into account while computing peers recommendations compared to the non weighted approach. The low performance of the schemes: Rogers and Tanimoto (4), Simple Matching (5), Sokal and Sneath (7) can be explained by the fact that these schemes take into consideration negative co-occurrence as explained in table I. If two peers do not have a file, a rating of 0 is assigned to this file. Considering that these two peers are similar if there are files that they both do not have, does not make them necessarily similar. It does not mean that they have the same interests. Indeed, the negative co-occurrence does not mean necessarily any resemblance or similarity in our context. However, an acceptable performance of the OchiaiII (6) scheme has been shown in the simulations although this scheme belongs to this category. On the other hand, similarity coefficients with no negative co-occurrence as described in table II, lead to better recommendations' accuracy and higher peers' satisfaction.

VII. Concluding Remarks

In this paper, we proposed a novel recommender framework for partially decentralized file sharing P2P systems. We investigated similarity metrics, that were proposed in other fields, and adapted them to file sharing P2P systems. We analyzed the impact of each similarity metric on the accuracy of the recommendations. Both weighted and non weighted approaches were investigated. In general, the weighted approaches achieve higher recommendation accuracy. Within the weighted approaches, similarity metrics that do not consider negative co-occurrence lead to better recommendation performance. Files' recommendations will, on one hand, increase users' satisfaction since they will receive recommendations on files that they prefer. On the other hand, they will help peers stay connected to the system to serve other peers in addition to increasing the peers' loyalty to the system.

References

[1] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl, "Item-Based Collaborative Filtering Recommendation Algorithms," in *International Conference on World Wide Web*, 2001, pp. 285–295.

[2] P. Massa and P. avesani, "Trust-aware Collaborative Filtering for Recommender Systems," in *International Conference on Cooperative Information Systems*, 2004.

[3] G. Ruffo, R. Schifanella, and E. Ghiringhello, "A Decentralized Recommendation System based on Self-Organizing Partnerships," in *IFIP Networking*, 2006, pp. 618–629.

[4] J. Wang, J. Pouwelse, R. L. Lagendijk, and M. J. T. Reinders, "Distributive Collaborative Filtering for Peer-to-Peer File Sharing Systems," in *ACM Symposium on Applied Computing*, 2006, pp. 1026–1030.

[5] J. Wang, J.A.Pouwelse, J. Fokker, A. de Vries, and M. Reinders, "Personalization on a Peer-to-Peer Television System," *Multimedia Tools and Applications*, vol. 36, pp. 89–113, 2008.

[6] E. Adar and B. Huberman, "Free Riding on Gnutella," HP, Tech. Rep., 2000.

[7] F. Lourenço, V. Lobo, and F. Baçao, "Binary-based Similarity Measures for Categorical Data and their Application in Self-Organizing Maps," in *JOCLAD*, 2004.

[8] J. Duarte, J. dos Santos, and L. Melo, "Comparison of similarity coefficients based on RAPD markers in the common bean," *Genetics and Molecular Biology*, vol. 22, no. 3, pp. 427–432, 1999.

[9] M. Lesot, M. Rifqi, and H.Benhadda, "Similarity Measures for Binary Numerical Data: a Survey," *International Journal on Knowledge Engineering and Soft Data Paradigms*, vol. 1, no. 1, pp. 63–84, 2009.

[10] L. Mekouar, Y. Iraqi, and R. Boutaba, "A Recommender Scheme for Peer-to-Peer Systems," in *IEEE International Symposium on Applications and the Internet International Workshop on Dependable and Sustainable Peer-to-Peer Systems*, 2008, pp. 197–200.

[11] K. Gummadi, R. J. Dunn, S. Saroiu, S. D. Gribble, H. M. Levy, and J. Zahorjan, "Measurement, Modeling, and analysis of a Peer-to-Peer File Sharing Workload," in *ACM Symposium on Operating Systems Principles*, 2003, pp. 314–329.

[12] "PeerSim," http://peersim.sourceforge.net/.

[13] G. Karypis, "Evaluation of Item-Based Top-N Recommendation Algorithms," in *International Conference on Information and knowledge Management*, 2001, pp. 247–254.