Journal of Computing and Security

January 2016, Volume 3, Number 1 (pp. 43–52)
http://www.jcomsec.org

Structural and Non-Structural Similarity Combination of Users in Social Networks

Azam Malekmohammad [a], Hadi Khosravi-Farsani [b,*]

[a] *Department of Computer Science, Safahan Institute of Higher Education, Isfahan, Iran.*
[b] *Department of Computer Engineering, Shahrekord University, Shahrekord, Iran.*

ARTICLE INFO.

Article history:
Received: 30 November 2015
Revised: 21 April 2016
Accepted: 12 October 2016
Published Online: 25 January 2017

Keywords:
Social Networks, Structural Similarity, Non-Structural Similarity.

ABSTRACT

Estimating similarity is expressed in many domains and sciences. For instance, data mining, web mining, clustering, search engines, ontology mapping and social networks require the definition and deployment of similarity. User similarity in social networks is one of the main problems and has many applications in this area. In this paper, a new method is introduced for combining structural and non-structural similarity between users in social networks. In the experimental section, structural similarity algorithms are combined with non-structural similarity algorithms through the proposed method. All experiments are implemented on some part of the Twitter dataset. Experimental results show that the precisions of all algorithms are increased with the proposed method.

1 Introduction

Social networks have become a part of everyone's life and are influencing their daily life. They have played a big role in how people interact with each other. The first social network was created in 1997 when the users could create personal profiles and add friends [1]. In later years, more social network communities came to rise in different areas. This technology rapidly spread across most of the countries. Social networks enable users to share information, ideas, interests, activities and life events. Examples of some popular social networking sites include Facebook, Twitter, LinkedIn, Google Plus+, etc.

A social network is structured as a graph where each vertex represents a person or an organization and the edge between two vertices indicates the friendship relationship, interaction, or collaboration among them. The social network's graph which has dynamic structure is changed over time by adding and removing nodes and edges.

In general, the social network's users have two structural and non-structural similarities. The structural similarity refers to the user's setting position in the network graph with the following features: vertex degree, number of triangles, clustering coefficient, eigenvector centrality, average shortest path length, and so on [2]. The structural similarities of the social network graph usually are expressed as neighboring matrices.

The non-structural or profiled similarity refers to the categorized data available in the user's profile. This profile includes the user's priorities, life events, and etc. As it may be clear, the categorized data is disorganized. Moreover, they are incomplete which means he/she does not feed data or makes some mistakes in doing so.

The similarity in social networks must be defined accurately based on predefined criteria. In this paper, the similarity is defined as "the users, who are sim-

* Corresponding author.

Email addresses: malekmohammad@safahan.ac.ir (A. Malekmohammad), khosravi@eng.sku.ac.ir (H. Khosravi-Farsani)

ISSN: 2322-4460 © 2016 JComSec. All rights reserved.

ilar in terms of predetermined criteria, are assigned an equal rank". This criterion can be a profile characteristic such as age, education, and place of education. Moreover, the user's network location in the corresponding graph can be used as a criterion for determining the similarity between users.

The similarity measure can be defined based on the underlying application. For example, in the recommender systems, the similarity between the user's answers can be used for similarity measure. In other words, a number of pre-determined questions are answered by the users. These answers create a vector of priorities for each user which is used by the user in similarity criteria. In social networks finding the similarity level of the people has vast applications (*e.g.* finding routes to expand news, ideas, theses, political views). Finding associations and pages interesting to the users or finding a friend of the same thought and ideas requires finding similar users. In all suggestive systems in order to find desired items the user needs to find similarity among his interests and the available items. The social networks are used in product commercialization and service marketing, based on the above-mentioned facts [3].

Determining the leaders and the followers in a social network is another application for similarity approximation in social networks [2].

Existing approaches in structural and profile similarity measures have some deficiency. Because of the low network connectivity between users and inaccessibility of profile information of users, the similarity between users will become hard to detect. So this paper tries to combine structural similarity and profile similarity to increase the accuracy of the measured similarity. In a social network, some users have profile similarity and some of them have structural similarity, so the use of an algorithm that supports structural and profile properties will produce better result.

The problem addressed in this paper is to propose a new method to estimate the users similarity in social networks. For this purpose, the paper is structured as follows. In Section 2, related works on the problem will be discussed and the approaches in estimating the users' similarity in social networks will be outlined. Also, structural similarity and non-structural similarity are defined correspondingly. The recommended method is on Section 3 and finally Section 4 is devoted to the evaluation of the proposed method.

2 Related Works

The similarity approximation algorithms are introduced to find the similarity level among the users, and constitute the following major groups:

- Algorithms that approximate the similarity by merely considering the structural features
- Algorithms that approximate the similarity by merely considering the non-structural features
- Algorithms that approximate the similarity by considering both structural and non-structural features

As the social networks expanded, together with the importance of their similarity measure, the first group of the mentioned algorithms were introduced and evolved. But it did not suffice since most of the users had no similarity as far as the structural similarity is concerned. However, in terms of the profiled similarity they were similar to each other. Consequently, the second group of the algorithms was introduced.

According to the homophily principle, the users who are similar based on profile measures are connected more in the structural sense [4]. This is the reason that in the recent years, the first and the second groups of the algorithms are combined to obtain better results. This has become possible since the structural and non-structural measures are two completely independent issues with detecting similarity. In this paper, a new approach for computing the similarity between users is proposed based on the structural and profile measures.

Many studies are available regarding similarity approximation where the structural features are adopted [3, 5–9], the same is true where non-structural features are adopted [10, 11]. However, very few studies have adopted the newly introduced "combined features".

Adoption of this new method is presented in [12], where in the structural section the NS and in the non-structural section, the OF algorithms are applied. The advantage of [12] in relation to the other few studies is that the users' profile structure is considered as a set of features where the multi-value features like education are supported as well. The majority voting method is adopted in [12] for inference of non-fed profile features. The obtained results are assessed on and compared with the directional YouTube and non-directional Facebook, social networks' graphs. There is no algorithm in [12] to measure the structural and non-structural similarity, but only the correlation of the structural and non-structural similarity is referred to.

Random walk with restart algorithm is introduced in [13] where the features of the available nodes and joints/connection are used to learn a grouping pattern in order for the nodes which have no connection with them to be assigned to positive or negative groups. The positive groups have the potential of being connected to the source node in future. To begin, the learning algorithm assigns a degree to the combination of two groups based on their features, allowing the Random walk algorithm to begin moving in the graph according to the assigned degrees with a tendency to move towards combinations with high degree. In this method the groups' and connections' features must be predetermined which makes its adoption to the real networks impossible.

One of the important applications of similarity approximation is clustering the users. The SA-cluster algorithm, which is introduced in [14] applies the users' structural and non-structural features for similarity approximation. The users within one cluster have the same grade. The SA-cluster algorithm does not consider the users' profile as grouped information but considers only one feature for the users' profile.

The structural similarity section of the proposed algorithm is P-Rank, SimRank and SRank algorithms. The first two is based on node's neighbors that are connected to the base node and calculate the similarity between two nodes based on the count of common neighbors. The SRank algorithm is based on the shortest distances between two nodes. In SRank all paths between two nodes is found using recursive functions and then the similarity degree is calculated.

2.1 SRank

The basic intuition of SRank [15] can be expressed as "two objects in a directed graph are considered similar if they are connected by a small-length path". More specifically, similarity between objects 'a' and 'b' (maybe different from similarity between 'b' and 'a') in a given graph is affected by the following two contradictory conditions:

- The number of shortest paths from a to b.
- The length of shortest paths from a to b.

To compute the user similarity based on SRank formula, first, we define access value. Let P_p be the $N \times N$ transition probability matrix with length p of a graph G. Access value from a to b is defined as:

$$H(a,b) = w_1 * P^1_{a,b} + \cdots + w_p * P^p_{a,b} + \ldots + w_{n-2} * P^{n-2}_{a,b} \tag{1}$$

Where w_i is the weight for all paths with length i. $P_p a, b$ is the probability of going from a to b with length p and is equal to the number of p-paths from a to b ($k_p(a,b)$) divided by the number of p-paths starting from a to other nodes ($k_p(a,x)$):

$$P^p_{a,b} = \frac{k_p(a,b)}{\sum_{\forall x \in G-\{a\}} k_p(a,x)} \tag{2}$$

The access values between different nodes of a given graph are approximately estimated by considering very few sentences of Equation (1). Moreover, constructing all of the different paths is very time-consuming. As a consequence, $H(a,b)$ is replaced by $H_s(a,b)$ and is defined as:

$$H_s(a,b) = w_1 * P^1_{a,b} + \cdots + w_s * P^s_{a,b} \quad 1 \leq s \leq n-2 \tag{3}$$

It is apparent that to obtain meaningful results, the weight of shorter paths must be higher than that of the longer paths:

$$w_p = 2^{s-p} \tag{4}$$

A straightforward approach to estimate similarity score between a and b is to normalize $H_s(a,b)$ with respect to H_{max} and H_{min} in the whole collection. In our experiment we used Equation (5) for the similarity score between a and b:

$$SRank_s(a,b) = \frac{H_s(a,b) - H_{min}}{H_{max} - H_{min}} \tag{5}$$

2.2 SimRank

The basic intuition of SimRank[5] can be expressed as "two objects in a directed graph are considered as similar if they are joined to similar neighbors". SimRank considers only in-neighbors to determine the similarity between users. The similarity between objects a and b is denoted by $s(a,b) \in [0,1]$. There is a recursive equation for $s(a,b)$: If $a = b$ then $s(a,b)$ is defined to be 1. Otherwise:

$$s(a,b) = \frac{C}{|I(a)||I(b)|} \sum_{i=1}^{|I(a)|} \sum_{j=1}^{|I(b)|} s(I_i(a), I_j(b)) \tag{6}$$

where C is a constant between 0 and 1. In most papers the value 0.8 is considered as C value. $I(a)$ and $I(b)$ are referred to the count of input neighbors of nodes 'a' and 'b' respectively. Input neighbors mean there is a link from the neighbor node to node 'a'. A slight technicality here is that either a or b may not have any in-neighbors. Since there is no way to infer any similarity between a and b, in this case, similarity value should be set to $s(a,b) = 0$. So, the summation in Equation (6) will be 0 when $I(a) = 0$ or $I(b) = 0$.

2.3 P-Rank

The basic recursive intuition of P-Rank[9] can be expressed as "two entities are similar if they are related to similar entities". PRank considers both in-neighbors and out-neighbors to determine the similarity between users.

If $a = b$ then $s(a,b)$ is defined to be 1. Otherwise:

$$s(a,b) = \lambda * \frac{C}{|I(a)||I(b)|} \sum_{i=1}^{|I(a)|} \sum_{j=1}^{|I(b)|} s(I_i(a), I_j(b)) + (1-\lambda) * \frac{C}{|O(a)||O(b)|} \sum_{i=1}^{|O(a)|} \sum_{j=1}^{|O(b)|} s(O_i(a), O_j(b)) \quad (7)$$

In Equation (7), $I(a)$ and $I(b)$ are referred to the count of input neighbors of node 'a' and 'b' respectively. $O(a)$ and $O(b)$ are referred to the count of output neighbors of node 'a' and 'b' respectively. Input neighbor means there is a link from the neighbor node to node 'a' and the output neighbor means there is a link from node 'a' to the neighbor node.

When $I(a)$ or $I(b) = 0$, the in-link part is invalidated and only the out-link direction takes into effect. Similarly, when $O(a)$ or $O(b) = 0$, only the similarity flowing from in-link parts are considered. If both $I(a)$ or $I(b) = 0$ and $O(a)$ or $O(b) = 0$, we define $s(a, b) = 0$. C represents the degree of attenuation in similarity propagation, and λ expresses the relative weight of similarity for in-link and out-link directions. When $\lambda = 0.5$, P-Rank well balances both in-link and out-link factors for measuring structural similarity. Like SimRank, in most papers, the value 0.8 is considered for C in P-Rank algorithm.

2.4 OF

In the proposed approach, OF algorithm is used in profile similarity section. OF algorithm is based on occurrence frequency as follows. Profile similarity between two users X and Y with the assumption of equality in profile feature's value will be one. Otherwise, it will be calculated as:

$$S(X,Y) = \frac{1}{1 + log\left(\frac{N}{f(X)}\right) * log\left(\frac{N}{f(Y)}\right)} \quad (8)$$

$F(x)$ is the number of occurrences of X in the whole user profile and $F(y)$ is the number of occurrences of Y in the whole user profile.

3 The Proposed Approach in Combining the Structural and Non-Structural Similarity Measures

The recommended approach is illustrated in Figure 1, which can be implemented for both the structural and profile similarity approximation algorithms. As it is observed in Figure 1, first, the most similarity level available for user X in relation to other available nodes in the graph who are not of X's friends are determined. If this value is bigger than zero, then node X certainly, has at least one connection with at least one node in the graph, otherwise $n = 0$. With respect to new users and the users who have the least connection with others, n is usually zero.

If n is bigger than zero, the number of users who are not among node X's friends and their structural similarity level with node X is equal or more than n are determined. If the number of the nodes exceeds a certain number, assessing all of them in test data is not a wise move. Therefore, this approach restricts the number of users who might have the most structural similarity with user X and if this number exceeds the allowable threshold, the recommendation is ignored and the profile similarity feature of user X with other users is considered; otherwise, the recommendation is made and for verification, it resorts to test data.

If a test data exists between user X and one of the users with the most structural similarity, it would indicate that a new combination is predicted for user X by this approach. In this study the threshold for the number of users with the most similarity with user X is considered to be 3. If the number of the users is less than 3, all of them will be recommended to user X, otherwise, the user profile similarity will be used in predicting the new combination.

In case of $n = 0$ the users' structural similarity cannot be applied in its new combined recommendation; since node X in such conditions has the least connection with other nodes or no connection at all, the users' profile similarity is more effective in predicting his/her new connections. Therefore, the most profile similarity level available for user X in relation to other available users is determined. This value will be considered as m and d will be the number of users that has profile similarity equal or greater than m with user X. If m is bigger than zero then, node X certainly has a significant similarity at least with one of the nodes in the graph. Here, a threshold limit of 10 is considered for the number of these nodes. If the number of the users whose profile similarity to X is equal to m is less than 10 all would be recommended to X and if it exceeds 10 no recommendation is made.

4 Evaluation

The Twitter social network is used in the evaluation of this study [16]. This data set consists of 21 million users and about 200 million friendships among the users; on average every user has 10 friends. Due to this high volume, a subset is extracted from the main data set by the Newman algorithm [7] and saved under a small graph which includes 2508 nodes and 8860 connections among the nodes. This small graph is divided into two parts named as Test data (for algorithm

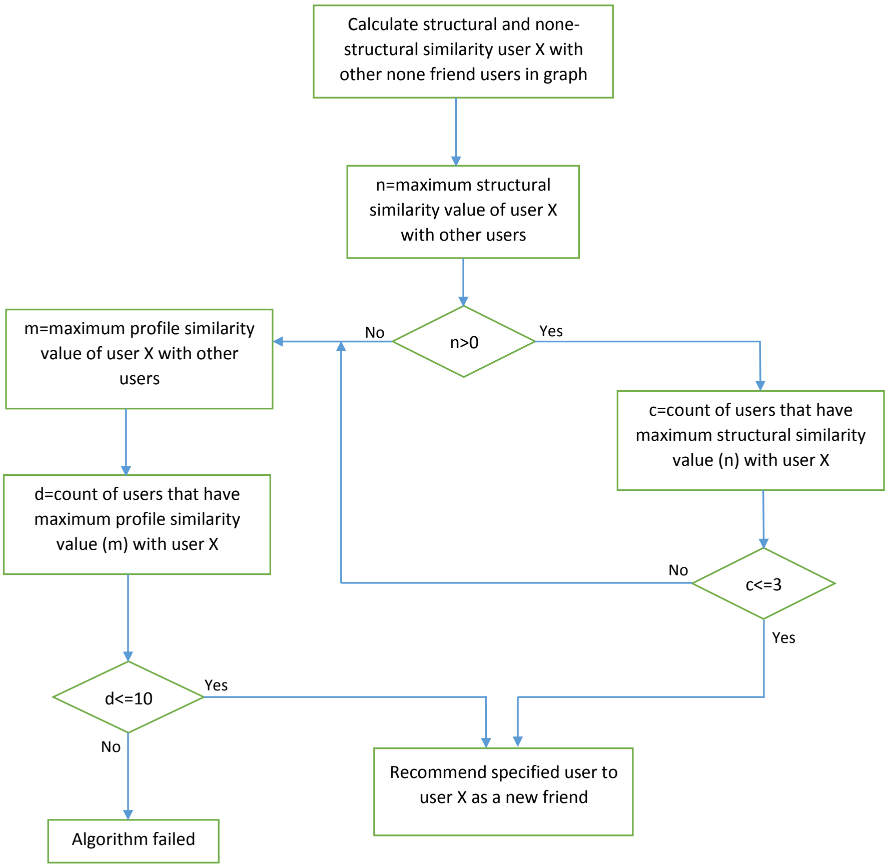
Calculate structural and none-structural similarity user X with other none friend users in graph
n=maximum structural similarity value of user X with other users
n>0
No
Yes
m=maximum profile similarity value of user X with other users
c=count of users that have maximum structural similarity value (n) with user X
d=count of users that have maximum profile similarity value (m) with user X
c<=3
No
Yes
d<=10
Yes
No
Recommend specified user to user X as a new friend
Algorithm failed

Figure 1. Proposed Method for Combining Structural and Non-Structural Similarity

implementation) and Train data in a random manner.

In this graph, one connection is eliminated from every node in a random manner and is placed in the test data section and the rest are placed in train data section. Algorithms are applied to the train data and seek to find one or more new friends with the most similarity and recommend it to the users. If there existed a prior relation between the user and the recommended friend in the test data, this algorithm has been able to make a correct prediction; otherwise, the results of its prediction are scarce. All nodes in this graph are subject to this process

4.1 Validation Metrics

For validating the results of the algorithms, we use three metrics.

(1) Precision: Also known as positive predictive value is the fraction of retrieved instances that are relevant. In the context of this paper, it means the percentage of suggested friends to all users that are correct and exist in the test data [17].

$$Precision = 100 * \frac{\text{number of correct found}}{\text{number of nodes}} \tag{9}$$

(2) Recall: Also known as sensitivity is the fraction of relevant instances that are retrieved. In the context of this paper, it means what percent of the potential friends who are in test data are suggested to users. High recall means that an algorithm returned most of the relevant results [17].

$$Recall = 100 * \frac{\text{number of correct found}}{\text{number of edges in test data}} \tag{10}$$

(3) F-Measure: Also known as F-score is a measure of test's accuracy. F-Measure is the harmonic mean of precision and recall. The F-Measure can be interpreted as a weighted average of the precision and recall, where F-Measure reaches its best value at 1 and worst score at 0 [17].

$$F - Measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{11}$$

4.2 Validation Results

In the first experiment, the three structural similarity methods of SRank, SimRank and P-Rank and the non-

Table 1. The Recommendation Results in Experiment1

Tests	Number of Edges in Test Data	Number of Edges in Train Data	Number of Correct Predictions			
			SimRank	P-Rank	SRank	OF
1	1772	7138	320	315	475	20
2	1760	7100	316	316	649	15
3	1763	7079	398	397	476	15
4	1750	7110	368	375	502	16
5	1775	6885	259	258	463	18

structural similarity method of OF are implemented on different samples extracted from the graph without applying the recommended algorithm.

In the second experiment, the three structural similarity methods of SRank, Sim-Rank and P-Rank and the non-structural similarity method of OF are combined and implemented on five samples extracted from the main graph. By comparing the results obtained from these two tests the success of the proposed algorithm is determined.

All experiments are implemented on one computer system with a 2.5 GHz and coretm5 with 4 GB main memory subject to windows 7 agent system in JAVA language.

4.2.1 Validation Results Without Applying the Recommended Algorithm

The results obtained from the initial tests on the five samples extracted from the graph are tabulated in Table 1. In both, the tests cycles for algorithms are run on the train data.

Description of the train data and test data of every test series are presented in the fifth and sixth columns of the subject test (*e.g.* in the first test in experiment1, after dividing the available connections in the graph on a random basis, 1772 connections are placed in the test data and 7138 connections are placed in the train data). These algorithms conduct similarity approximation for all the users in the graph and recommend a new connection to them. Test data is used to determine the correctness of the recommended connection by the algorithm. If the recommended connection is contained in the test data the recommendation by the algorithm is correct; otherwise, it is not correct. The numbers of correct recommendations by SimRank, P-Rank and SRank algorithms are presented in columns 4, 3 and 2, respectively. The same, regarding OF algorithm, is presented in the first column. With respect to the number of correct recommendations of each algorithm, the precision of each algorithm is obtained for experiment1 and is shown in Table 2 and Figure 2.

Table 2. Precision Results in Experiment1

Test	SimRank	P-Rank	SRank	OF
test1	12.75	12.55	18.93	0.8
test2	12.59	12.59	18.7	0.6
test3	15.89	15.82	18.97	0.6
test4	14.67	14.95	20.01	0.63
test5	10.32	10.28	18.46	0.71

Table 3. Recall Results in Experiment1

Test	SimRank	P-Rank	SRank	OF
test1	18.53	18.3	27.58	1.16
test2	17.95	17.95	26.64	0.85
test3	22.57	22.51	30	0.85
test4	21.02	21.42	28.68	0.91
test5	14.75	14.7	26.36	1.01

Table 4. F-Measure Results in Experiment1

Test	SimRank	P-Rank	SRank	OF
test1	15.1	14.88	22.45	0.94
test2	14.79	14.79	21.97	0.7
test3	18.64	18.58	23.24	0.7
test4	17.28	17.6	23.57	0.74
test5	12.14	12.09	21.73	0.83

The results of recall obtained in experiment1 are shown in Table 3 and Figure 3. the results of F-measure for the same test are shown in Table 4 and Figure 4. The F-measure results indicate that the SRank algorithm which performs based on the short routes between two nodes, in comparison with other algorithms which are based on neighborhood nodes is of higher accuracy. It becomes evident that OF algorithm alone has a poor performance in predicting new connections.

Table 5. The Recommendation Results in Experiment2

Tests	Number of Edges in Test Data	Number of Edges in Train Data	Number of Correct Predictions		
			SimRank & OF	P-Rank & OF	SRank & OF
6	1772	7138	426	425	687
7	1760	7100	421	421	698
8	1763	7079	412	413	706
9	1750	7110	410	409	709
10	1775	6885	398	398	689

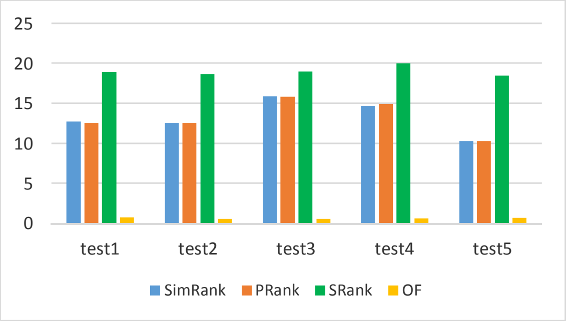

Figure 2. Precision Results in Experiment1

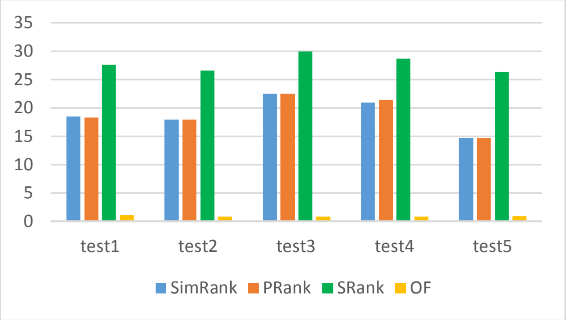

Figure 3. Recall Results in Experiment1

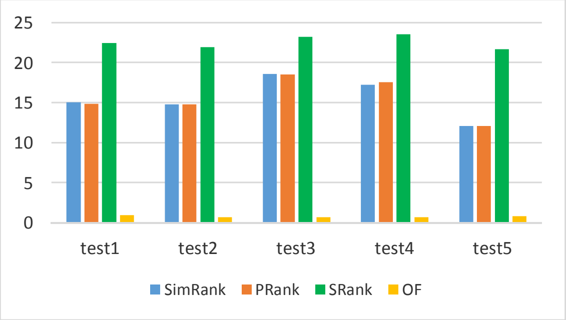

Figure 4. F-Measure Results in Experiment1

Table 6. Precision Results in Experiment2

Test	SimRank	P-Rank	SRank
test6	16.95	16.94	27.39
test7	16.78	16.78	27.83
test8	16.42	16.46	28.14
test9	16.34	16.3	28.26
test10	15.86	15.89	27.47

Table 7. Recall Results in Experiment2

Test	SimRank	P-Rank	SRank
test6	24.73	24.68	39.89
test7	23.92	23.92	39.56
test8	23.36	23.42	40.04
test9	23.42	23.37	40.51
test10	22.67	22.67	38.81

4.2.2 The Results of Applying the Recommended Algorithm

The results obtained from the next tests on five samples extracted from the graph are tabulated in Table 5. In this test, the structural similarity algorithms are combined with the OF non-structural similarity algorithm through the proposed method and applied as a single algorithm on train data. The number of correct predictions of the algorithm with respect to the major changes applied to the users is shown in Table 5. The precision of the second test results, with respect to Table 2 content, are shown in Table 6 and Figure 5.

The results of recall obtained in experiment2 are shown in Table 7 and Figure 6, and the results of F-measure for the same experiment, are shown in Table 8 and Figure 7.

Here it is obvious that the combination of algorithm SRank with OF, in comparison with algorithms based on neighborhood nodes is of higher accuracy. The

Table 8. F-Measure Results in Experiment2

Test	SimRank	P-Rank	SRank
test6	20.11	20.09	32.47
test7	19.72	19.72	32.7
test8	19.28	19.33	33.05
test9	19.24	19.2	33.29
test10	18.66	18.68	32.16

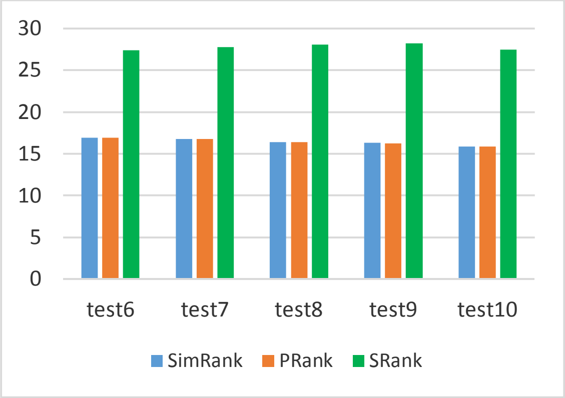

Figure 5. Precision Results in Experiment2

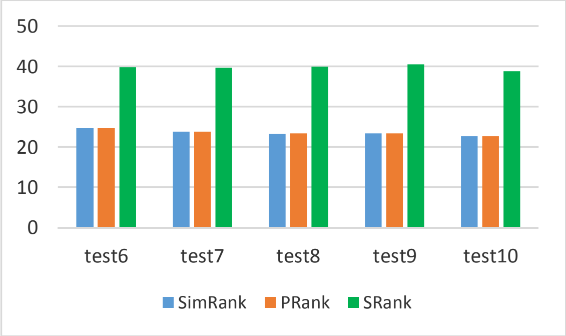

Figure 6. Recall Results in Experiment2

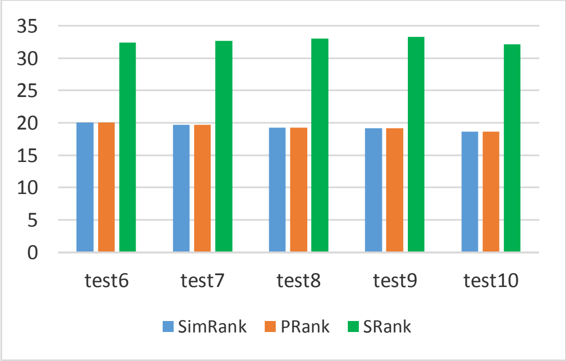

Figure 7. F-Measure Results in Experiment2

precision, recall and F-measure prediction of each algorithm show about 1.5 fold increases, according to the results obtained from both experiments.

5 Conclusion

In this paper a new algorithm is introduced which combines the structural and non-structural similarity algorithms. To implement this algorithm the SRank, SimRank, and P-Rank algorithms are used for determining the structural similarity and the OF algorithm used for determining the non-structural similarity. The experiments are conducted on a subset of the Twitter social network dataset in order to approximate user similarity in social networks and predict the new connections among them. By applying the Newman's assembly/recognition algorithm, a small graph is extracted from the main data set and some connections on this graph are eliminated on a random basis. The objective of the algorithms is predicting the eliminated connections. The results obtained from the two experiments revealed that the accuracy rate of this newly introduced algorithm in predicting new connections has improved by 1.5 fold. Another finding is that the structural similarity based on the short route is more accurate in approximating similarity and prediction of the new connections, compared with the structural similarity algorithms based on node neighborhood. According to the proposed algorithm, the structural similarity based on the short route can be combined with the algorithms of non-structural similarity in a better manner compared with the one based on neighborhood nodes.

The experiments here indicate that the SRank algorithm outperforms others with regard to accuracy in similarity approximation.

Improving this algorithm regarding the accuracy in similarity approximation would constitute the theme of the upcoming article. Another topic will be the implementation of this newly introduced algorithm on data sets with multi-profile fields.

References

[1] Won Kim, Ok-Ran Jeong, and Sang-Won Lee. On social Web sites. *Information Systems*, 35 (2):215 – 236, 2010. ISSN 0306-4379. doi: http://dx.doi.org/10.1016/j.is.2009.08.003. URL `http://www.sciencedirect.com/science/article/pii/S0306437909000866`. Special Section: Context-Oriented Information Integration.

[2] M. Z. Shafiq, M. U. Ilyas, A. X. Liu, and H. Radha. Identifying Leaders and Followers in Online Social Networks. *IEEE Journal on Selected Areas in*

Communications, 31(9):618–628, September 2013. ISSN 0733-8716. doi: 10.1109/JSAC.2013.SUP.0513054.

[3] M. Zhang, Z. He, H. Hu, and W. Wang. E-rank: A Structural-Based Similarity Measure in Social Networks. In *2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*, volume 1, pages 415–422, Dec 2012. doi: 10.1109/WI-IAT.2012.111.

[4] Miller McPherson, Lynn Smith-Lovin, and James M. Cook. Birds of a Feather: Homophily in Social Networks. *Annual Review of Sociology*, 27: 415–444, 2001. ISSN 03600572, 15452115. URL `http://www.jstor.org/stable/2678628`.

[5] Glen Jeh and Jennifer Widom. SimRank: A Measure of Structural-context Similarity. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '02, pages 538–543, New York, NY, USA, 2002. ACM. ISBN 1-58113-567-X. doi: 10.1145/775047.775126. URL `http://doi.acm.org/10.1145/775047.775126`.

[6] Lada A Adamic and Eytan Adar. Friends and neighbors on the Web. *Social Networks*, 25(3): 211 – 230, 2003. ISSN 0378-8733. doi: http://dx.doi.org/10.1016/S0378-8733(03)00009-1. URL `http://www.sciencedirect.com/science/article/pii/S0378873303000091`.

[7] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Phys. Rev. E*, 69:026113, Feb 2004. doi: 10.1103/PhysRevE.69.026113. URL `http://link.aps.org/doi/10.1103/PhysRevE.69.026113`.

[8] Ellen Spertus, Mehran Sahami, and Orkut Buyukkokten. Evaluating Similarity Measures: A Large-scale Study in the Orkut Social Network. In *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, KDD '05, pages 678–684, New York, NY, USA, 2005. ACM. ISBN 1-59593-135-X. doi: 10.1145/1081870.1081956. URL `http://doi.acm.org/10.1145/1081870.1081956`.

[9] Peixiang Zhao, Jiawei Han, and Yizhou Sun. P-Rank: A Comprehensive Structural Similarity Measure over Information Networks. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09, pages 553–562, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-512-3. doi: 10.1145/1645953.1646025. URL `http://doi.acm.org/10.1145/1645953.1646025`.

[10] Shyam Boriah, Varun Chandola, and Vipin Kumar. *Similarity Measures for Categorical Data: A Comparative Evaluation*, pages 243–254. 2008. doi: 10.1137/1.9781611972788.22. URL `http://epubs.siam.org/doi/abs/10.1137/1.9781611972788.22`.

[11] Prantik Bhattacharyya, Ankush Garg, and Shyhtsun Felix Wu. Analysis of user keyword similarity in online social networks. *Social Network Analysis and Mining*, 1(3):143–158, 2011. ISSN 1869-5469. doi: 10.1007/s13278-010-0006-4. URL `http://dx.doi.org/10.1007/s13278-010-0006-4`.

[12] C. G. Akcora, B. Carminati, and E. Ferrari. Network and profile based measures for user similarities on social networks. In *2011 IEEE International Conference on Information Reuse Integration*, pages 292–298, Aug 2011. doi: 10.1109/IRI.2011.6009562.

[13] Lars Backstrom and Jure Leskovec. Supervised Random Walks: Predicting and Recommending Links in Social Networks. In *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, WSDM '11, pages 635–644, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0493-1. doi: 10.1145/1935826.1935914. URL `http://doi.acm.org/10.1145/1935826.1935914`.

[14] Yang Zhou, Hong Cheng, and Jeffrey Xu Yu. Graph Clustering Based on Structural/Attribute Similarities. *Proc. VLDB Endow.*, 2(1):718–729, August 2009. ISSN 2150-8097. doi: 10.14778/1687627.1687709. URL `http://dx.doi.org/10.14778/1687627.1687709`.

[15] Hadi Khosravi-Farsani, Mohammadali Nematbakhsh, and Georg Lausen. SRank: Shortest paths as distance between nodes of a graph with application to RDF clustering. *Journal of Information Science*, 39(2):198–210, 2013. doi: 10.1177/0165551512463994. URL `http://dx.doi.org/10.1177/0165551512463994`.

[16] Rui Li, Shengjie Wang, Hongbo Deng, Rui Wang, and Kevin Chen-Chuan Chang. Towards social user profiling: unified and discriminative influence model for inferring home locations. In *KDD*, pages 1023–1031, 2012.

[17] Stephen Robertson. *Evaluation in Information Retrieval*, pages 81–92. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001. ISBN 978-3-540-45368-0. doi: 10.1007/3-540-45368-7_4. URL `http://dx.doi.org/10.1007/3-540-45368-7_4`.

Azam MalekMohammad received her B.Sc from Ashrafi Isfahani University in Computer Engineering in 2010. Also she received M.Sc from Safahan University in 2015. Her subject in M.Sc thesis was "determining structural and none-structural similarity between users in social networks". She had a paper in international conference on web mining in 2015. Now, She is working in Kasra Co.

Hadi Khosravi-Farsani received his B.Sc, M.Sc and Ph.D from University of Isfahan in Computer Engineering in 2004, 2007, and 2012 respectively. He has joined to Shahrekord University as academic member from 2012 and has taught Object-Oriented Programming, Web programming, and Database Concepts. He is head of computer engineering Department from 2015. At the same time, He has working for Tiida Co. as a product manager from 2012 and has produced many web-based applications for university requirements. Content Management System(CMS), Learning Management System(LMS), Conference Management System and Laboratory Management are some of his important applications. His current research is Data-oriented application, Big Data, and web-based applications.