

Fast and Accurate Computation of Role Similarity via Vertex Centrality

Longjie Li¹(✉), Lvjian Qian¹, Victor E. Lee², Mingwei Leng³,
Mei Chen^{1,4}, and Xiaoyun Chen¹

¹ School of Information Science and Engineering,
Lanzhou University, Lanzhou 730000, China
{ljli,mchen,chenxy}@lzu.edu.cn

² Department of Mathematics and Computer Science,
John Carroll University, University Heights, OH 44118, USA

³ School of Mathematics and Computer Science,
Shangrao Normal University, Shangrao 334001, China

⁴ School of Electronic and Information Engineering,
Lanhou Jiaotong University, Lanzhou 730070, China

Abstract. There is growing evidence that vertex similarity based on structural context is the basis of many link mining applications in complex networks. As a special case of vertex similarity, role similarity which measures the similarity between two vertices according to their roles in a network can facilitate the search for peer vertices. In *RoleSim*, graph automorphism is encapsulated into the role similarity measure. As a real-valued role similarity, *RoleSim* shows good interpretative power in experiments. However, *RoleSim* is not sufficient for some applications since it is very time-consuming and may assign unreasonable similarities in some cases. In this paper, we present *CentSim*, a novel role similarity metric which obeys all axiomatic properties for role similarity. *CentSim* can quickly calculate the role similarity between any two vertices by directly comparing their corresponding centralities. The experimental results demonstrate that *CentSim* achieves best performance in terms of efficiency and effectiveness compared with the state-of-the-art.

Keywords: Complex network · Vertex similarity · Role similarity · Vertex centrality · Similarity metric

1 Introduction

Nowadays, networked data, *e.g.*, social network and web page, is proliferating and attracting a growing interest among researchers. In sociology, individuals

L. Li—L. Li was supported by the Fundamental Research Funds for the Central Universities (No. lzujbky-2014-47).

M. Chen—M. Chen was supported by the Gansu Provincial Natural Science Fund (No. 145RJZA194) and the Fundamental Research Fund for the Gansu Universities (No. 214151).

are often assigned “social roles”, such as a father, a doctor, or a professor. In the past, role studies have primarily been the interest of sociologists on offline social networks [1, 17]. Recent studies have found that roles also appear in many other type of networks, such as biological networks [13], web graphs [14], and technological networks [15]. On the one hand, role discovery is indeed an important task for general graph mining and exploratory analysis since it is useful in many real applications [20–22]. On the other hand, measuring role-based similarity between any two vertices is also a key question in studying the roles in a network system [7]. One reason is role similarity can help to predict vertex functionality within their domains. For instance, in a protein-protein interaction network, proteins with similar roles usually serve similar metabolic functions. Thus, if the function of one protein is known, all other proteins having the similar role would be predicted to have similar function [5].

Despite its significance, the problem of role similarity has received little attention. From the viewpoint of the network, automorphic vertices have equivalent surroundings and hence share the same role. In [7], graph automorphism is encapsulated into the role similarity measure: *two automorphically equivalent vertices share the same role and have maximal role similarity*. Take the network shown in Fig. 1a as an example. Clearly, vertices b and c are automorphically equivalent, thus they share the same role and should have the maximal role similarity. Although vertices d and e are not automorphically equivalent, they have the very similar surroundings and hence should have higher role similarity. To estimate how role-similar two vertices are, a real-valued role similarity measure, called RoleSim [7], was proposed. For vertex-pairs (b, c) and (d, e) , RoleSim can successfully assign their similarities, *i.e.*, $RoleSim(b, c) = 1$ and $RoleSim(d, e) = 0.589$. For other two vertex-pairs (a, b) and (a, e) , $RoleSim(a, b) < RoleSim(a, e)$. This result is acceptable and reasonable.

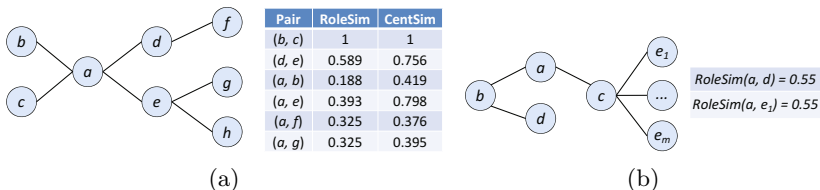


Fig. 1. Two example networks. The damping factor of RoleSim is set to be 0.1 and the initialization of RoleSim is ALL-1 scheme [7]. The Degree, PageRank and Closeness centralities are used in CentSim, and their weights are equally set to 1.

However, RoleSim assigns the same similarity score to vertex-pairs (a, f) and (a, g) in Fig. 1a. That seems to be unreasonable. Furthermore, in Fig. 1b, RoleSim always thinks that vertex-pairs (a, d) and (a, e_1) have the same role similarity regardless of the number of neighbors of vertex c . Although this situation should not be deemed as a failure of RoleSim, there is clearly room to improve its accuracy and sensitivity. In addition, the very serious problem in RoleSim is it is a time-consuming method. Its time complexity is $O(kN^2d')$ where k is the

number of iterations, N is the number of vertices in a network, and d' is the average of $(d_u \times d_v) \times \log(d_u \times d_v)$ of all vertex-pairs (u, v) . d_u is the degree of vertex u . In [8], a scalable algorithm for RoleSim, namely IcebergRoleSim, was presented to speed the computation of RoleSim while the cost is to prune the vertex-pairs whose similarities are lower than a given threshold.

In this paper, we propose CentSim (centrality-based similarity measure), a new real-valued role similarity measure to quickly and accurately compute the role similarity of any vertex-pairs. In CentSim, we employ the centralities of vertices to calculate their role similarities. When measuring two vertices' role similarity, CentSim just compares several of their centralities, while RoleSim investigates all neighbor-pairs of the two vertices. Thus, CentSim can quickly calculate any vertex-pairs' role similarities. Furthermore, CentSim obeys all the axiomatic role similarity properties [7] and hence is an admissible role similarity metric. To show the performance of CentSim, we review the network shown in Fig. 1a. Vertices b, c have the same role and d, e have the very similar surroundings, CentSim can assign reasonable similarities to (b, c) and (d, e) , respectively. That is $CentSim(b, c) = 1$ and $CentSim(d, e) = 0.756$. For vertex-pairs (a, f) and (a, g) , $CentSim(a, f) = 0.376$ and $CentSim(a, g) = 0.395$. For the network shown in Fig. 1b, CentSim always assigns a smaller similarity to (a, d) than that to (a, e_1) for any value of $m > 1$. Compared to the similarities assigned by RoleSim, these results are more reasonable.

The rest of this paper is organized as follows. Section 2 gives a brief introduction of related work, and Section 3 contains some preliminaries for this work. In Section 4, we detail the proposed role similarity measure and prove it is an admissible role similarity metric. Section 5 demonstrates the experimental results. Finally, the conclusion of this paper is presented in Section 6.

2 Related Work

To date, many link-based similarities have been proposed. Among them, SimRank [6] is a well-known one, which is based on the intuition that *two vertices are similar if they are linked by similar vertices*. The computation of SimRank is iterative. In each iteration, SimRank updates the similarity score between two different vertices according to the average similarity of all their neighbor pairs in the previous iteration. The idea of SimRank seems to be solid and elegant; however, it may assign inaccurate or even counter-intuitive similarity scores [12] as well as undesirably introduces the “zero-similarity” issue [25, 26]. With the help of *maximal weighted matching* of neighbor pairs, MatchSim [12] overcomes the counter-intuitive results of SimRank. In order to remedy the “zero-similarity” issue, SimRank* [25] introduces a new strategy to find more paths that are largely overlooked by SimRank. Similarly, E-Rank [26] deals with the meetings of two vertices that walk along any length paths and also solves the “zero-similar” issue.

PageSim [11] is a quite different link-based similarity measure from SimRank. Motivated by the propagating mechanism of PageRank [19] and simultaneously employing the PageRank scores as vertices' features, PageSim propagates PageRank score of each vertex to other vertices via links, and then represents each

vertex by a feature vector. The PageSim score of one vertex-pair is derived by comparing their feature vectors.

Role similarity, which measures the similarity of vertices based on their roles, is a special case in the link-based similarity problem. RoleSim [7], an admissible role similarity metric, was proposed to evaluate how role-similar two vertices are. Two main properties make RoleSim a role similarity metric. The first is when updating the similarity between two different vertices, RoleSim adopts the maximal weighted matching between their neighbors. And the second one is the initialization of RoleSim is admissible.

3 Preliminaries

In this section, we give the necessary background and notations before we discuss role similarity further.

3.1 Role Similarity Properties

A social network or other complex network is defined as an undirected graph $G(V, E)$ where V is the vertex set and E represents the edge set. For a given vertex u in graph G , the set of its neighbors is denoted as $N(u)$ and the degree of u is the number of its neighbors, denoted as d_u , $d_u = |N(u)|$.

Given a graph $G(V, E)$ and two vertices $u, v \in V$, an *automorphism* of G is a *permutation* σ of V such that $(u, v) \in E$ iff $(\sigma(u), \sigma(v)) \in E$. If $u = \sigma(v)$, then vertices u and v are *automorphically equivalent*, denoted as $u \equiv v$.

To theoretically depict the role similarity measure, Jin *et al.* [7] formulated a series of axiomatic properties that all role similarity measures should obey.

Definition 1 (Axiomatic Role Similarity Properties). *Let $G(V, E)$ be a graph and $s(u, v)$ be the similarity score between any two vertices $u, v \in V$. Five axiomatic properties of role similarity are developed as follows:*

- 1) *Range:* $s(u, v) \in [0, 1]$.
- 2) *Symmetry:* $s(u, v) = s(v, u)$.
- 3) *Automorphism confirmation:* If $u \equiv v$, $s(u, v) = 1$.
- 4) *Transitive similarity:* If $u \equiv v, x \equiv y$, then $s(u, x) = s(u, y) = s(v, x) = s(v, y)$.
- 5) *Triangle inequality:* $d(u, x) \leq d(u, v) + d(v, x)$, where $d(u, v) = 1 - s(u, v)$.

If $s(u, v)$ obeys the first four properties, it is an **admissible role similarity measure**. If $s(u, v)$ satisfies all five properties, it is called an **admissible role similarity metric**.

3.2 Centrality

Centrality is a general measure of vertex activity in a network and can be calculated by several metrics. The most popular ones are *degree*, *closeness*, *betweenness* and *eigenvector centrality* [18]. These measures determine the relative importance of a vertex within a network, particularly a social network. In a social network, vertices with larger values of centrality measures are powerful vertices and occupy the critical positions [4].

The degree centrality represents the local importance of a vertex. Generally, a vertex with higher degree is inclined to have a greater ability of local influence than others, or to be closer to the center of a network. Closeness measures the reachability of a vertex to other vertices. Formally, it is the average length of all shortest paths from a given vertex to all others in a network. Higher closeness value of a vertex indicates more vertices can be reached with shorter paths, which fits the human intuition of “centrally located.” The betweenness of a vertex can commonly be interpreted as the frequency that this vertex lies on the shortest paths between any two vertices. A vertex with high betweenness usually occupies a critical position which connects two different regions and controls the information flow between different communities. Eigenvector centrality measures the influence or importance of a vertex. The basic idea of eigenvector centrality is that the influence of a vertex is recursively defined by the influence of its neighbors. *PageRank* [19] can be treated as a variant of eigenvector centrality.

4 CentSim: A Novel Role Similarity Metric

In this section, we describe the proposed new role similarity measure, CentSim.

4.1 Definition of CentSim

The basic idea of our similarity measure comes from two aspects. The first one is the role of a vertex is deeply influenced by its position in a network. And the second one is centrality is a general measure of how the position of a vertex is within a network [18]. Thus, centrality can be elected as a favorable tool to evaluate role similarity. Consequently, in CentSim, we employ vertex’s centralities to calculate their role similarities. The formal computation of CentSim is given in Definition 2.

Definition 2 (CentSim). *Given a graph $G(V,E)$ and two vertices $u, v \in V$, the CentSim score between u and v is defined as:*

$$\text{CentSim}(u, v) = \frac{\sum_{i=1}^l w_i \theta_i(u, v)}{\sum_{i=1}^l w_i} \quad (1)$$

where l is the number of different centralities adopted in CentSim. Coefficient $w_i > 0$ is the weight of centrality c_i . $\theta_i(u, v)$ is defined as:

$$\theta_i(u, v) = \frac{\min(c_i(u), c_i(v))}{\max(c_i(u), c_i(v))} \quad (2)$$

where $c_i(u)$ is the value of the centrality c_i of u . In Equation 2, we define $\frac{0}{0} = 1$.

From Definition 2, we can see that the core of the computation of CentSim score is to compare the centrality values of vertices. Suppose the centrality values of vertices are obtained in advance. Clearly, the CentSim score between any two vertices can be computed straightforwardly. In our implementation, three centralities, *i.e.*, PageRank, Degree and Closeness, are employed. In default, we set the weights of the three centralities equally to 1.

4.2 Admissibility of CentSim

Theorem 1 (Admissibility). *CentSim is an admissible role similarity metric.*

To prove Theorem 1, we can separately prove that CentSim obeys each of the five axiomatic role similarity properties listed in Definition 1. Trivially, CentSim holds true for the Range (property 1) and Symmetry (property 2). For the Transitive similarity (property 4), Jin *et al.* [7] proved that it is implied by the Triangle inequality property. Therefore, in the following, we only need to prove that CentSim satisfies the Automorphism confirmation (property 3) and Triangle inequality (property 5).

Lemma 2. *For any two vertices u, v in graph G , if $u \equiv v$ then $c_i(u) = c_i(v)$.*

Actually, two automorphically equivalent vertices are identical with respect to all graph theoretic properties and hence have the same centrality score on every possible measure (see chapter 12 in [24]).

Proof of Automorphism Confirmation. Since $u \equiv v$, in the light of Lemma 2, we get $\theta_i(u, v) = \frac{\min(c_i(u), c_i(v))}{\max(c_i(u), c_i(v))} = \frac{c_i(u)}{c_i(v)} = 1$. So that

$$CentSim(u, v) = \frac{\sum_{i=1}^l w_i \theta_i(u, v)}{\sum_{i=1}^l w_i} = \frac{\sum_{i=1}^l w_i}{\sum_{i=1}^l w_i} = 1. \quad \square$$

Proof of Triangle Inequality. Given any vertices x, y and z in G , we get

$$\begin{aligned} d(x, y) + d(y, z) - d(x, z) &= 1 + CentSim(x, z) - CentSim(x, y) - CentSim(y, z) \\ &= 1 + \frac{\sum_{i=1}^l w_i \theta_i(x, z)}{\sum_{i=1}^l w_i} - \frac{\sum_{i=1}^l w_i \theta_i(x, y)}{\sum_{i=1}^l w_i} - \frac{\sum_{i=1}^l w_i \theta_i(y, z)}{\sum_{i=1}^l w_i} \\ &= \frac{\sum_{i=1}^l w_i (1 + \theta_i(x, z) - \theta_i(x, y) - \theta_i(y, z))}{\sum_{i=1}^l w_i} \end{aligned}$$

Let $c_i(x) = a$, $c_i(z) = b$ and $c_i(y) = c$, then

$$\begin{aligned} 1 + \theta_i(x, z) - \theta_i(x, y) - \theta_i(y, z) &= 1 + \frac{\min(a, b)}{\max(a, b)} - \frac{\min(a, c)}{\max(a, c)} - \frac{\min(b, c)}{\max(b, c)} \\ &= \frac{|a - c|}{\max(a, c)} + \frac{|b - c|}{\max(b, c)} - \frac{|a - b|}{\max(a, b)} \end{aligned}$$

Thus, to prove CentSim satisfies triangle inequality, it suffices to show that Inequality 3 holds true.

$$\frac{|a - b|}{\max(a, b)} \leq \frac{|a - c|}{\max(a, c)} + \frac{|b - c|}{\max(b, c)} \quad (3)$$

If a , b and c are positive numbers, the proof of Inequality 3 can be found in [23]. If one or two or all of a , b and c are 0s, Inequality 3 also holds true, since we defined that $\frac{0}{0} = 1$ in Definition 2. Therefore, CentSim satisfies the Triangle inequality property. \square

In conclusion, CentSim obeys all the axiomatic role similarity properties. That is to say, CentSim is an admissible role similarity metric.

4.3 Complexity of CentSim

Suppose the centrality values of vertices are given in advance. Obviously, the time complexity of CentSim is $O(lN^2)$ ($l \ll N$) where N is the number of vertices in graph G and l is the number of centralities used in CentSim. In fact, l is a constant when CentSim is implemented, thus the time complexity of CentSim is reduced to $O(N^2)$. To obtain the similarity of all vertex-pairs, we need to compute $N(N-1)/2$ similarity scores and record lN centrality values. Therefore, the space complexity of CentSim is $O(lN + N^2)$.

5 Experiments

In this section, we experimentally study the performance of CentSim in terms of efficiency and effectiveness. Five baselines are SimRank [6], SimRank* [25], MatchSim [12], PageSim [11] and RoleSim [7].

5.1 Experimental Setup and Datasets

All experiments are conducted on a machine with AMD Opteron 8347 4 core CPU and 16GB DDR2 memory. The operating system is Suse Linux Enterprise Server 10 SP2. CentSim and five baselines are implemented in C++, while the scores of PageRank, Degree and Closeness centrality are computed by the NetworkX¹ package of Python. In [25], the authors presented an algorithm for computing SimRank* by means of *fine-grained memoization*, namely memo-gSR*. However, in this paper, we only implement the naive algorithm of SimRank*. We set the damping factors $C = 0.8$ for both SimRank and SimRank*, and $\beta = 0.1$ for RoleSim. The initialization of RoleSim is ALL-1 [7].

For impartial comparison of similarity measures, we utilize four real-world datasets from varying fields as benchmarks, which are PGP [16], Yeast [2], Enron [9] and DBLP². The DBLP is a co-author network derived from 7-year publications (2006-2012) in conferences of SIGMOD, VLDB, ICDE, KDD, ICDM, and

¹ <http://networkx.github.io/>

² <http://dblp.uni-trier.de/~ley/db/>

Table 1. Statistics of the largest connected components of the four networks. N : number of vertex; M : number of edge; $\langle k \rangle$: average degree of vertices; k_{max} : maximal degree of vertices; $\langle d \rangle$: average shortest distance of all vertex-pairs.

	PGP	Yeast	Enron	DBLP
N	10680	2224	33696	5890
M	24316	7049	180811	19845
$\langle k \rangle$	4.554	6.339	10.732	6.739
k_{max}	205	66	1383	157
$\langle d \rangle$	7.486	4.377	4.025	5.782

Table 2. Running time of all measures on four benchmarks (unit: second).

	PGP	Yeast	Enron	DBLP
SimRank	1939	200	108121	984
SimRank*	849	57	20395	327
MatchSim	4614	921	273625	2604
PageSim	4395	93	422835	1271
RoleSim	4890	1003	326754	2698
CentSim	91	5	981	28
Centralities	486	13	7204	156

SDM. Each network is treated as an undirected unweighted graph and pruned into its largest connected component. The statistics are listed in Table 1.

5.2 Comparison of Time Performance

This section compares the time performance of CentSim with the five baselines. We perform each measure on the four benchmark datasets to compute the similarities of all vertex-pairs and then count the running time. The results of time are listed in Table 2. From Table 2, we can clearly see that CentSim outperforms the others on all benchmarks. This achievement of CentSim is due to its straightforward computation of similarity. The last row in Table 2 gives the total time of computing the scores of PageRank, Degree and Closeness centrality. As shown in Table 2, even counting the time of computing centralities, CentSim still costs the least time compared with the baselines. Therefore, our CentSim is more efficient than the state-of-the-art on assigning the similarity scores for vertex-pairs.

5.3 Comparison of Accuracy Performance

For the time performance, it is easy to evaluate by tracking the total running time. However, evaluating the performance of accuracy is quite hard, since it is difficult to identify a benchmark in which the real roles of vertices are identified or the role similarities of vertices are known. To delineate roles, two alternatives are utilized in this work: (1) as in [7], we use K -shell [3] as a proxy; (2) we adopt the roles of vertices discovered by the method proposed in [27] as ground-truth.

To quantitatively evaluate the performance of accuracy of a role similarity measure, two criteria are utilized in this paper.

The first criterion comes from the following idea. That is, the higher role similarity score two vertices have, the more likely they are within the same shell or share the same role. To formulate this idea, we compute the **fraction** of top ranked vertex-pairs that are within the same shells or share the same roles.

The accuracy performance evaluated by the fraction based on the two alternatives are shown in Fig. 2 and 3, respectively. Obviously, CentSim achieves the

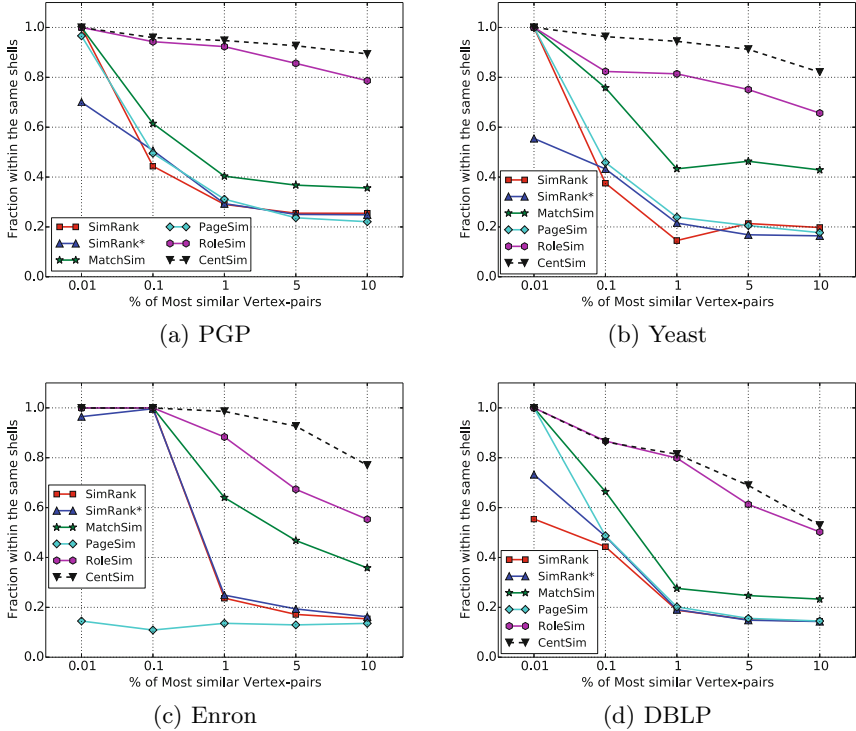


Fig. 2. Fraction of top ranked vertex-pairs that within the same shells

best performance on accuracy while RoleSim obtains the second best. In Fig. 2, on both PGP and Yeast, all measures, except SimRank*, do well for the top 0.01% vertex-pairs. However, CentSim markedly outperforms the baselines when more top-ranked vertex-pairs are considered. On Enron, PageSim unexpectedly achieves the very bad results, while others do well for the top 0.01% and 0.1% vertex-pairs. And furthermore, when the range is expanded, CentSim still does well, while the performance of both SimRank and SimRank* decline significantly. In our viewpoint, two reasons cause the poor accuracy of PageSim on Enron: one is the variety of vertex's degree is large, and the other is that the feature propagating mechanism makes PageSim to assign high similarity scores to vertex-pairs cross-shells. For the top 0.01% vertex-pairs ranked by PageSim, the maximum, average and variance of the difference of degree of vertex-pair are 1382, 12.118 and 1375.711, respectively. On DBLP, CentSim, RoleSim, MatchSim and PageSim do very well for the top 0.01% vertex-pairs; CentSim and RoleSim do well for the top 0.1% and 1% vertex-pairs. But CentSim shows better performance on accuracy than RoleSim for the top 5% and 10% vertex-pairs. In Fig. 3, the results are similar to those of Fig. 2. Due to lack of space, the details are omitted. In one word, experimental results in Fig. 2 and 3 indicate that more similar vertex-pairs ranked by CentSim are more likely to be within the same roles.

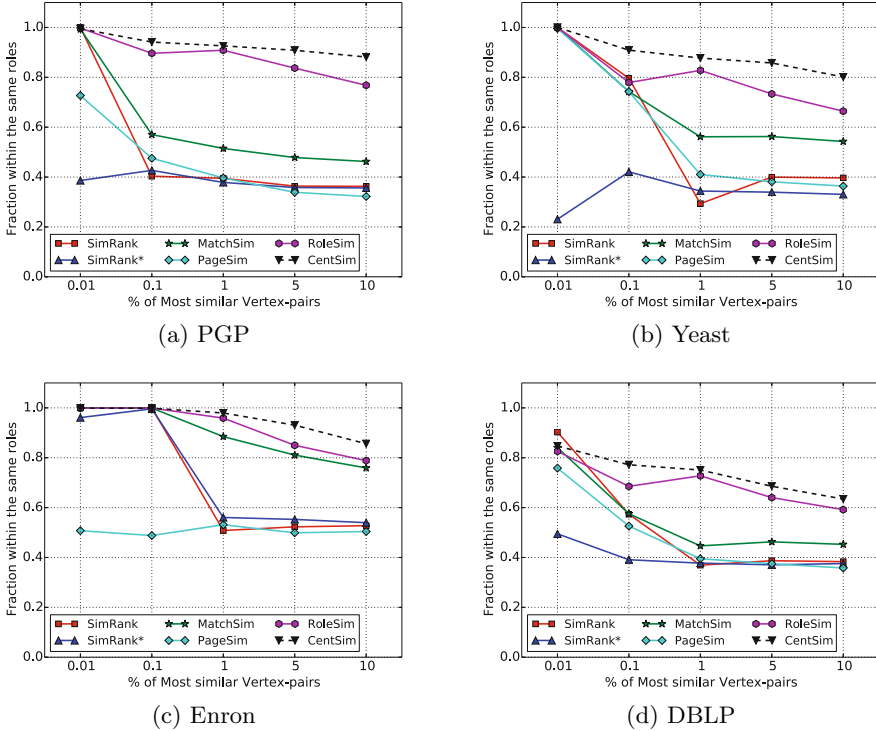


Fig. 3. Fraction of top ranked vertex-pairs that within the same roles

Table 3. Accuracies of similarity measures measured by auc statistic (ground-truth: K -shell). Each number is obtained by averaging over 100 independent realizations.

	PGP	Yeast	Enron	DBLP
SimRank	0.485	0.448	0.491	0.551
SimRank*	0.568	0.466	0.795	0.635
MatchSim	0.527	0.680	0.635	0.618
PageSim	0.428	0.374	0.498	0.489
RoleSim	0.822	0.864	0.877	0.827
CentSim	0.823	0.872	0.933	0.776

Table 4. Accuracies of similarity measures measured by auc statistic (ground-truth: role [27]). Each number is obtained by averaging over 100 independent realizations.

	PGP	Yeast	Enron	DBLP
SimRank	0.494	0.478	0.500	0.508
SimRank*	0.580	0.491	0.590	0.608
MatchSim	0.531	0.639	0.588	0.504
PageSim	0.444	0.407	0.499	0.424
RoleSim	0.760	0.755	0.720	0.627
CentSim	0.820	0.801	0.795	0.699

The basic opinion of the second criterion is that the role similarity of two vertices within-role (or shell) should be bigger than that of the other vertices cross-role (or shell). So, we group all vertex-pairs into two parts: within-role pairs, P^w , and cross-role pairs, P^c . To quantify the accuracy of similarity measures, we use the **AUC statistic**. It can be interpreted as the probability that a randomly chosen within-role pair (a pair in P^w) is given a higher role similarity score than a

randomly chosen cross-role pair (a pair in P^c). In the implementation, among n independent comparisons, if there are n' cases that the within-role pair has the higher score, and n'' cases that the within-role pair and the cross-role pair have the same score, as in the research of link prediction [10], we define AUC in Eq. 4.

$$AUC = \frac{n' + 0.5 \times n''}{n} \quad (4)$$

The accuracy results measured by AUC are shown in Table 3 and 4, respectively. Generally speaking, CentSim can give overall better accuracy than the baselines. After CentSim, RoleSim performs the next best, while the others, particularly PageSim and SimRank, perform far worse. These results show that CentSim has high probability to assign large role similarity scores for vertex-pairs within-role.

In summary, the above experiments conducted in Section 5.2 and 5.3 demonstrate that CentSim is not only more efficient than the state-of-the-art but also outperforms them in accuracy. Therefore, we can conclude that the framework of measuring role similarities of vertices by comparing their centrality scores is competitive. Consequently, CentSim is competent to the task of role similarity.

6 Conclusion

In this paper, we proposed a novel and qualified similarity measure, namely CentSim, to quickly and accurately assign the role similarity between any two vertices of a network. We observe that the role of a vertex is related to its position in a network and centrality generally measures the position of a vertex in a network. Motivated by these two aspects, CentSim computes the role similarity between two vertices by means of comparing their corresponding centralities. Importantly, CentSim is an admissible role similarity metric since it obeys all the axiomatic role similarity properties.

We experimentally evaluate the performance of CentSim in terms of efficiency and effectiveness compared with SimRank, SimRank*, MatchSim, PageSim and RoleSim on four real-world datasets. The experimental results demonstrate that CentSim achieved overall best performance on both time and accuracy compared with the state-of-the-art. Thus, CentSim is a qualified role similarity metric.

References

1. Anderson, C.J., Wasserman, S., Faust, K.: Building stochastic blockmodels. *Social Networks* **14**(1), 137–161 (1992)
2. Bu, D., Zhao, Y., Cai, L., Xue, H., Zhu, X., Lu, H., Zhang, J., Sun, S., Ling, L., Zhang, N., Li, G., Chen, R.: Topological structure analysis of the protein-protein interaction network in budding yeast. *Nucleic Acids Research* **31**, 2443–2450 (2003)
3. Carmi, S., Havlin, S., Kirkpatrick, S., Shavitt, Y., Shir, E.: A model of internet topology using k-shell decomposition. *PNAS* **104**(27), 11150–11154 (2007)
4. Freeman, L.C.: Centrality in social networks: Conceptual clarification. *Social Networks* **1**(3), 215–239 (1979)

5. Holme, P., Huss, M.: Role-similarity based functional prediction in networked systems: application to the yeast proteome. *Journal of the Royal Society Interface* **2**(4), 327–333 (2005)
6. Jeh, G., Widom, J.: Simrank: a measure of structural-context similarity. In: *SIGKDD 2002*, pp. 538–543. ACM (2002)
7. Jin, R., Lee, V.E., Hong, H.: Axiomatic ranking of network role similarity. In: *SIGKDD 2011*, pp. 922–930. ACM (2011)
8. Jin, R., Lee, V.E., Li, L.: Scalable and axiomatic ranking of network role similarity. *TKDD* **8**(1), 3 (2014)
9. Klimt, B., Yang, Y.: Introducing the enron corpus. In: *CEAS 2004*
10. Lü, L., Zhou, T.: Link prediction in complex networks: A survey. *Physica A* **390**(6), 11501170 (2011)
11. Lin, Z., King, I., Lyu, M.R.: Pagesim: A novel link-based similarity measure for the world wide web. In: *WI 2006*, pp. 687–693. IEEE (2006)
12. Lin, Z., Lyu, M.R., King, I.: Matchsim: a novel neighbor-based similarity measure with maximum neighborhood matching. In: *CIKM 2009*, pp. 1613–1616. ACM (2009)
13. Luczkovich, J., Borgatti, S.P., Johnson, J.C., Everett, M.G.: Defining and measuring trophic role similarity in food webs using regular equivalence. *Journal of Theoretical Biology* **220**(3), 303–321 (2003)
14. Ma, H., King, I., Lyu, M.R.: Mining web graphs for recommendations. *TKDE* **24**(6), 1051–1064 (2012)
15. Mahadevan, P., Krioukov, D., Fomenkov, M., Dimitropoulos, X., Claffy, K., Vahdat, A.: The internet as-level topology: Three data sources and one definitive metric. *ACM SIGCOMM Computer Communication Review* **36**(1), 17–26 (2006)
16. Ná, M.B., Satorras, R.P., Guílera, A.D., Arenas, A.: Models of social networks based on social distance attachment. *Physical Review E* **70**, 056122 (2004)
17. Nowicki, K., Snijders, T.A.B.: Estimation and prediction for stochastic blockstructures. *Journal of the American Statistical Association* **96**(455), 1077–1087 (2001)
18. Oliveira, M.: Gama, J.A.: An overview of social network analysis. *Wiley Interdisciplinary Reviews. Data Mining and Knowledge Discovery* **2**(2), 99–115 (2012)
19. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: Bringing order to the web. In: *WWW 1998*, pp. 161–172. ACM (1998)
20. Rossi, R.A., Ahmed, N.K.: Role discovery in networks. arXiv preprint [arXiv:1405.7134](https://arxiv.org/abs/1405.7134) (2014)
21. Rossi, R.A., Gallagher, B., Neville, J., Henderson, K.: Modeling dynamic behavior in large evolving graphs. In: *WSDM 2013*, pp. 667–676. ACM (2013)
22. Ruan, Y., Parthasarathy, S.: Simultaneous detection of communities and roles from large networks. In: *COSN 2014*, pp. 203–214. ACM (2014)
23. Tiakas, E., Papadopoulos, A., Nanopoulos, A., Manolopoulos, Y., Stojanovic, D., Djordjevic-Kajan, S.: Searching for similar trajectories in spatial networks. *Journal of Systems and Software* **82**(5), 772–788 (2009)
24. Wasserman, S., Faust, K.: *Social Network Analysis: Methods and Applications*. Cambridge University Press (1994)
25. Yu, W., Lin, X., Zhang, W., Chang, L., Pei, J.: More is simpler: Effectively and efficiently assessing node-pair similarities based on hyperlinks. *PVLDB* **7**(1), 13–24 (2013)
26. Zhang, M., He, Z., Hu, H., Wang, W.: E-rank: A structural-based similarity measure in social networks. In: *WI-IAT 2012*, pp. 415–422. IEEE (2012)
27. Zhuge, H., Zhang, J.: Topological centrality and its e-science applications. *Journal of the American Society for Information Science and Technology* **61**(9), 1824–1841 (2010)