SimRank and Its Variants in Academic Literature Data: Measures and Evaluation

Masoud Reyhani Hamedani
Department of Computer and Software
Hanyang University, Seoul, Korea
masoud@agape.hanyang.ac.kr

Sang-Wook Kim
Department of Computer and Software
Hanyang University, Seoul, Korea
wook@agape.hanyang.ac.kr

ABSTRACT

SimRank is a well-known link-based similarity measure that can be applied on a citation graph to compute similarity of academic literature data. The intuition behind SimRank is that *two objects are similar if they are referenced by similar objects*. SimRank has attracted a growing interest in the areas of data mining and information retrieval recently. Despite of the current success of SimRank, it has some problems that negatively affect its effectiveness in similarity computation. In this paper, we discuss the *three existing problems* of SimRank, present SimRank *variants* that have been proposed to solve those problems, and evaluate the effectiveness of SimRank and its variants in similarity computation for academic literature data by conducting extensive experiments on a *real-world* dataset.

CCS Concepts

•Human-centered computing → Social networking sites; *Social networks;* **•Information systems → Retrieval effectiveness;**

Keywords

SimRank, SimRank Problems, SimRank Variants, Similarity, Academic Literature Data

1. INTRODUCTION

Academic papers are one of primary sources for researchers to share information and knowledge [14]. Recently, the number of academic papers has been growing exponentially. While researchers suffered from a shortage of information in the past, they have begun to suffer from an excessiveness of information these days [4]. Therefore, there is a strong need on computational tools known as academic literature search engines such as Google Scholar[1], CiteSeer[2], and Microsoft Academic Search[3] to alleviate this problem [14][4]. One of the most challenging issues in academic literature search engines is computing the similarity of two papers, which is utilized to find those papers relevant to a paper in question [14]. A dataset of academic papers can be represented as a citation graph where nodes correspond to papers and edges do citation relationships among papers [14]. Figure 1 shows a sample citation graph. Link-based similarity measures such as SimRank [8] can be applied to compute the similarity of academic papers by exploiting the citation graph [14][10].

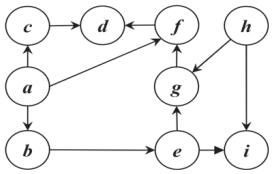

Figure 1: A sample citation graph.

Recently, SimRank and its variants have attracted a growing interest in the areas of data mining and information retrieval [23][19][5]. The philosophy behind SimRank is that *two objects are similar if they are referenced by similar objects* [8]. Based on this philosophy, the similarity score between a pair of papers in a citation graph is *iteratively* computed as the *average* of similarity scores among *all possible pairs* of papers citing them. SimRank has been successfully applied in different applications such as clustering [2], citation analysis [14][7], query rewriting [1], vertex join query [23][20], range search [5], and k-nearest neighbor search [5][9]. There are two representations for SimRank:

(1) ***Iterative Form*** [8]. For a given citation graph $G = (V, E)$ where V represents the set of papers and $E \in V \times V$ is a set of citation relationships among papers, the SimRank score of a paper-pair (a, b), $S(a, b)$, is computed as follows:

$$S(a,b) = \begin{cases} 1, & a = b \\ \frac{C}{|I_a||I_b|} \sum_{i \in I_a} \sum_{j \in I_b} S(i,j), & a \neq b \end{cases} \quad (1)$$

where I_a denotes a set of papers *directly* citing paper a, $|I_a|$ is the size of I_a, and $C \in (0, 1)$ is a damping factor. If $I_a = \emptyset$ or $I_b = \emptyset$, $S(a, b) = 0$. Equation (1) is a recursive formula

[1]http://scholar.google.com
[2]http://citeseerx.ist.psu.edu
[3]http://academic.research.microsoft.com

SAC 2016, April 04-08, 2016, Pisa, Italy

DOI: http://dx.doi.org/10.1145/2851613.2851811

Table 1: Similarity scores of some paper-pairs in our sample citation graph.

Paper-Pairs	SimRank	rvs-SimRank	P-Rank	PSimRank	C-Rank	MatchSim	SimRank*
(b,c)	0.8	0	0.459	0.4	0.8	1.0	0.064
(g,i)	0.4	0	0.249	0.4	0.8	1.0	0.039
(c,e)	0	0	0	0	0	0	0.038
(e,h)	0	0.4	0.249	0	0	0	0
(b,g)	0	0	0	0	0	0	0.026

started by $S_0(a,b) = 1$ if $a = b$; $S_0(a,b) = 0$ otherwise. For successive iterations $k = 1, 2, ...$, we have

$$S_k(a,b) = \begin{cases} 1, & a = b \\ \frac{C}{|I_a||I_b|} \sum_{i \in I_a} \sum_{j \in I_b} S_{k-1}(i,j), & a \neq b \end{cases} \quad (2)$$

In similarity computation, SimRank does not only consider the papers directly citing a and b but also considers the papers *indirectly* citing them.

(2) ***Matrix Form*** [21]. Lets $S \in \mathbb{R}^{(|V| \times |V|)}$ be a *similarity matrix* whose entry $[S]_{a,b}$ denotes $S(a,b)$; then, SimRank scores are computed as follows:

$$S = C \cdot (Q^T \cdot S \cdot Q) \vee I \quad (3)$$

where $Q_{|V| \times |V|}$ is a *column normalized* adjacency matrix whose entry $[Q]_{a,b} = 1/|I_b|$ if a cites b; $[Q]_{a,b} = 0$ otherwise. Q^T is a transpose of Q and $I_{|V| \times |V|}$ is an identity matrix. $\vee$ is a disjunction operator selecting the maximum value between its operations. The iterative computation starts with $S_0 = I$. Equation (3) is a *non-linear* recursive matrix form, which computes *exact* SimRank scores; however, it is hard to be computed because each time we have to figure out the maximum entry in the disjunction operator [9].

Another matrix form for SimRank proposed in references [10] and [7] is as follows:

$$S = C \cdot (Q^T \cdot S \cdot Q) + (1 - C) \cdot I \quad (4)$$

where term $(1 - C) \cdot I$ guarantees that main diagonal entries in S are always maximum (i.e., a paper is most similar to itself). Since Equation (4) is a *linear* recursive matrix form, it is easier to be computed than Equation (3); however, it *does not* provide the exact SimRank scores because the diagonal entries in S are not equal to one. Nevertheless, this approximation does not make a tangible effect on the similarity-based ranking [9].

In spite of the current success of SimRank, it has some problems caused by its philosophy, which negatively affect the effectiveness of SimRank in similarity computation. In the literature, significant efforts have been devoted to improve the effectiveness of SimRank by providing solutions to its existing problems. In this paper, (1) we discuss the *existing* problems of SimRank in term of *effectiveness*[4] by providing some samples; (2) we present SimRank variants that have been proposed to solve its problems; (3) we evaluate the effectiveness of SimRank and SimRank variants in similarity computation by conducting extensive experiments on a *real-world* dataset of academic papers. To the best of our knowledge, this is the first work that extensively covers and discusses all aforementioned issues together.

[4]SimRank also has problem in term of efficiency [5][7][9][20].

The rest of the paper is organized as follows. Section 2 discusses the existing problems of SimRank. Section 3 presents SimRank variants. Section 4 explains our experimental setup and analyzes the results of our experiments. Section 5 summarizes the paper.

2. SIMRANK PROBLEMS

In this section, we discuss the existing problems of SimRank in term of effectiveness by providing samples. Consider the sample citation graph in Figure 1. Table 1 shows the similarity scores of some paper-pairs in our sample citation graph computed by different similarity measures. Three existing problems of SimRank are as follows.

First, to compute $S(a,b)$, SimRank considers the papers citing a and b while neglects those papers cited by them. Therefore, when $I_a = \emptyset$, $I_b = \emptyset$, or no common papers directly (indirectly) cite both of them, $S(a,b) = 0$ even if they directly (indirectly) cite some common papers [22]. As an example, consider paper-pair (e,h) in Figure 1. Since $I_h = \emptyset$, $S(e,h)$ tends to be zero as shown in Table 1. However, papers e and h can be considered likely similar because both of them cite papers g and i. We refer to this issue as an *in-links consideration problem*.

Second, SimRank computes the similarity score of a paper-pair (a,b) iteratively as the *average* of similarity scores among *all* possible pairs of papers between I_a and I_b. Therefore, the SimRank score of a paper-pair commonly cited by a *large* number of papers tends be lower than that of another paper-pair commonly cited by a *small* number of papers [11][18][3]. As an example, consider paper-pairs (b,c) and (g,i) in Figure 1. The number of common papers citing b and c (i.e., only a) is less than that for g and i (i.e., e and h). Therefore, the similarity score of (b,c) should not be higher than that of (g,i). In spite of this fact, the SimRank score of (b,c) (i.e., 0.8) is higher than that of (g,i) (i.e., 0.4) as shown in Table 1. Consider another sample [18][3] when a large number of papers (i.e., n papers) cite both papers a and b. Also, there is not any citation relationships between those n papers. $S(a,b)$ is computed as $\frac{C}{n}$, which decreases and converges to zero as the number of common papers citing both a and b increases. We refer to this counter-intuitive issue as a *pairwise normalization problem*.

Third, SimRank regards papers a and b as similar if some paths *only with equal length* exist from a common paper to both of them [19]. As an example, consider paper-pair (d,e) in Figure 1. There are two paths with length *two* from paper a to paper d; one path via paper c and one path via paper f. Also, there is a path with length *two* from paper a to paper e via paper b. Therefore, SimRank regards papers d

and e as similar. Now consider paper-pair (c, e); there are *no* paths with equal length from any common papers (i.e., a) to them. There is a path with length *two* from paper a to paper e, and there is a path with length *one* from paper a to paper c. Therefore, $S(c, e)$ tends to be zero as shown in Table 1. However, papers c and e can be considered likely similar because at least some paths exist from the common paper a to both of them. In other words, SimRank exploits the graph topology level-by-level [17]; the similarity score between two papers belonging to different levels tends to be zero. We refer to this counter-intuitive issue as a *level-wise computation problem*.

3. SIMRANK VARIANTS

In the literature, significant efforts have been devoted to improve the effectiveness of SimRank in similarity computation. In this section, we present SimRank variants that have been proposed to solve its existing problems.

P-Rank [22] solves the *in-links consideration problem*. To compute the similarity score of a paper-pair (a, b), P-Rank exploits papers citing them as well as those papers cited by them on *each iteration* as follows. The average of similarity scores among all possible pairs of papers *citing* papers a and b is computed. Also, the average of similarity scores among all possible pairs of papers *cited* by them is computed as well. Then, a *weighted linear combination* of the two obtained scores is regarded as the P-Rank score of (a, b), $R(a, b)$. Furthermore, P-Rank provides a unified formulation so that other link-based similarity measures such as SimRank and rvs-SimRank can be regarded as its special cases [22]. In contrast to SimRank, rvs-SimRank computes the similarity between a pair of papers by considering *only* the papers cited by them. $R(a, b) = 1$ if $a = b$; otherwise

$$R(a,b) = \alpha \cdot \frac{C}{|I_a||I_b|}\Sigma_{i\in I_a}\Sigma_{j\in I_b}R(i,j) + (1-\alpha)\cdot\frac{C}{|O_a||O_b|}\Sigma_{i\in O_a}\Sigma_{j\in O_b}R(i,j) \quad (5)$$

where O_a is a set of papers *directly* cited by a and $\alpha \in [0, 1]$ is a weighting parameter. If $\alpha = 1$, Equation (5) denotes the SimRank formula; if $\alpha = 0$, it does the rvs-SimRank formula. The iterative computation started by $R_0(a, b) = 1$ if $a = b$; $R_0(a, b) = 0$ otherwise. As shown in Table 1, the P-Rank score of (e, h) is not zero in contrast to its SimRank score. We should notice that rvs-SimRank has the same problem as SimRank since the rvs-SimRank score between two papers with *no* common papers cited by them tends to be zero such as (b, c) as shown in Table 1.

PSimRank [3], C-Rank [18], and MatchSim [11] solve the *pairwise normalization problem*. In contrast to SimRank, PSimRank does not compute the similarity score of a paper-pair (a, b) by considering the all possible pairs of papers between I_a and I_b. Instead, on *each iteration*, the similarity score of (a, b) is regarded as the summation of Jaccard Coefficient score between I_a and I_b, the average of similarity scores among all possible pairs of papers between $I_a - I_b$ and I_b, and the average of similarity scores among all possible pairs of papers between $I_b - I_a$ and I_a. In this way, PSimRank allows random walkers in a and b to meet up in one step with higher probability (i.e., $\frac{|I_a \cap I_b|}{|I_a \cup I_b|}$) than SimRank. Informally, the intuition behind this measure is that the papers in $I_a \cap I_b$ are more informative in similarity computation than other papers in I_a and I_b. PSimRank score of a paper-pair (a, b), $PS(a, b)$, is computed as follows. If $a = b$, $PS(a, b) = 1$; if $I_a = \emptyset$ or $I_b = \emptyset$, $PS(a, b) = 0$; otherwise

$$PS(a,b) = C\cdot\Big(\frac{|I_a\cap I_b|}{|I_a\cup I_b|} + \frac{|I_a - I_b|}{|I_a\cup I_b|}\cdot\frac{\Sigma_{i\in I_a - I_b}\Sigma_{j\in I_b}PS(i,j)}{|I_a - I_b||I_b|} + \frac{|I_b - I_a|}{|I_a\cup I_b|}\cdot\frac{\Sigma_{i\in I_b - I_a}\Sigma_{j\in I_a}PS(i,j)}{|I_b - I_a||I_a|}\Big) \quad (6)$$

on the initial step, $PS_0(a, b) = 1$ if $a = b$; $PS_0(a, b) = 0$ otherwise.

C-Rank [18] also solves the pairwise normalization problem as in PSimRank. If $a = b$, $CR(a, b) = 1$; if $I_a = \emptyset$ or $I_b = \emptyset$, $CR(a, b) = 0$; otherwise

$$CR(a,b) = C\cdot\Big(\frac{|I_a\cap I_b|}{|I_a\cup I_b|} + \frac{\Sigma_{i\in I_a - I_b}\Sigma_{j\in I_b}CR(i,j)}{|I_a - I_b||I_b|} + \frac{\Sigma_{i\in I_b - I_a}\Sigma_{j\in I_a}CR(i,j)}{|I_b - I_a||I_a|}\Big) \quad (7)$$

on the initial step, $CR_0(a, b) = 1$ if $a = b$; $CR_0(a, b) = 0$ otherwise.

MatchSim [11] proposes a different solution to the pairwise normalization problem. MatchSim score of a paper-pair (a, b), $MS(a, b)$, is computed as the *average* of similarity scores of all paper-pairs in a *maximum matching* between I_a and I_b. MatchSim constructs a weighted bipartite graph $G'_{a,b} = (I_a, I_b, E', w)$ where $E' = \{(a', b') | a' \in I_a, b' \in I_b\}$ and $w(a', b')$ is set to the MatchSim score of paper-pair (a', b'). $MS(a, b)$ is computed as follows:

$$MS(a,b) = \frac{\widehat{W}(a,b)}{max(|I_a|,|I_b|)} \quad (8)$$

where $\widehat{W}(a, b)$ denotes the weight of a maximum matching between I_a and I_b in $G'_{a,b}$ computed as the summation of similarity scores between any possible pairs of papers belonging to the maximum matching. MatchSim iterative computation starts with $MS_0(a, b) = 1$ if $a = b$; $MS_0(a, b) = 0$ otherwise.

As shown in Table 1, PSimRank, C-Rank, and MatchSim do not regard the similarity score of (b, c) as higher than that of (g, i) in contrast to SimRank.

SimRank* [19] solves the *level-wise computation problem* by considering those paths neglected by SimRank during similarity computation. Although SimRank* considers more paths in similarity computation than SimRank, it does not suffer from increased computational cost because it employs an efficient matrix form as follows:

$$\widehat{S} = \frac{C}{2}\cdot(Q^T\cdot\widehat{S} + \widehat{S}\cdot Q) + (1-C)\cdot I \quad (9)$$

The iterative computation starts with $\widehat{S}_0 = I$. Since $\widehat{S}$ is a symmetric matrix, $\widehat{S}.Q$ is identical to the transpose of $Q^T.\widehat{S}$; Equation (9) requires only one matrix multiplication in comparison with Equations (3) and (4), which need two

matrix multiplications. As shown in Table 1, the SimRank* score of (c, e) is not zero in contrast to its SimRank score[5].

4. EXPERIMENTAL EVALUATION

In this section, we evaluate the effectiveness of SimRank and its variants in similarity computation with a real-world dataset of academic papers. Section 4.1 describes our experimental setup. Section 4.2 presents and analyzes the results.

4.1 Experimental Setup

All our experiments were performed on an Intel machine equipped with four 3.4 GHz i5-4670 CPUs, 32 GB RAM, and a 64-bit Fedora Core 20 operating system. All required codes were implemented with Java based on Open JDK 1.7.0.

We employed a real-world dataset of academic papers by crawling information of 1,071,793 papers from DBLP[6]. Also, we crawled their related citation information from MS Academic Search. The evaluation of similarity measures is difficult without performing extensive user studies [8][14]. However, the evaluation based on user studies is quite expensive and time-consuming [14]. Therefore, we constructed our *ground truth sets* based on a famous data mining textbook [6] where, as in user studies, relevant papers to the research topics discussed in each chapter have been categorized by experts (i.e., the authors of the book) in the bibliographic section of the chapter. We selected eleven research topics as our ground truth sets each of which contains its related papers. Our dataset consists of those papers in the areas of data mining and databases published in 2006 and before in consistent with [6]. The total number of papers in our dataset and ground truth sets are 22,179 and 143, respectively. Also, the average number of citations is 6.85.

In order to evaluate the effectiveness, we utilize MAP (mean average precision), precision, recall [13], and PRES (patent retrieval evaluation score) [12] as evaluation metrics. PRES considers the rank of retrieved relevant papers in a result set based on the recall; $PRES = 1 - \frac{\frac{\Sigma r_i}{n} - \frac{n+1}{n}}{N_{max}}$ where n is the number of relevant papers and N_{max} is the size of a result set. r_i is a rank of the i^{th} relevant paper in the result set. For x numbers of relevant papers, which are *not* retrieved, a rank is assigned by starting from $(N_{max} + n - x + 1)$ since those x relevant papers are located somewhere after N_{max} in the result set.

We consider top 10 results in our evaluations. We use every single paper in a ground truth set g as a query and compute the average precision (AP), precision, recall, and PRES for that query; if a paper in the result set belongs to the ground truth set g, it is labeled as *relevant*, otherwise *irrelevant*. In this way, we compute MAP, precision, recall, and PRES over *all* the queries in the ground truth set g. Finally, we calculate the average of MAP, precision, recall, and PRES over all eleven ground truth sets as the final results.

We implement SimRank* by using a CSR (compressed spare row) format [15]. We implement SimRank, rvs-SimRank, and P-Rank by using CSC (compressed spare column) format [15] based on the matrix form in Equation (3) since it computes the exact SimRank values. By utilizing compressed formats, the time and space complexity of a matrix multiplication become $O(|V|m)$ and $O(|V| + m)$, respectively; m denotes the number of non-zero entries in the adjacency matrix. We set the damping factor C as 0.8 for SimRank, rvs-SimRank, and P-Rank by following [22] and the relative weight α as 0.5 for P-Rank. In the cases of SimRank*, PSimRank, and C-Rank, we set the value of C as 0.6, 0.4, and 0.8 by following [19], [3], and [18], respectively.

[5]SimRank* solves the existing problem in RWR as well. More details can be found in [19].

[6]http://www.informatik.uni-trier.de

4.2 Results and Analyses

In this section, we analyze and compare the effectiveness of SimRank and its variants in similarity computation for academic papers. We compute similarity by applying each measure in ten iterations.

First, we compare the effectiveness of rvs-SimRank and SimRank in similarity computation. rvs-SimRank and SimRank assigns a similarity score to 237,105,577 and 89,310,415 paper-pairs in ten iterations, receptively. Although rvs-SimRank assigns a similarity score to more paper-pairs, SimRank outperforms rvs-SimRank on all iterations in terms of MAP, precision, PRES, and recall as shown in Figure 2.

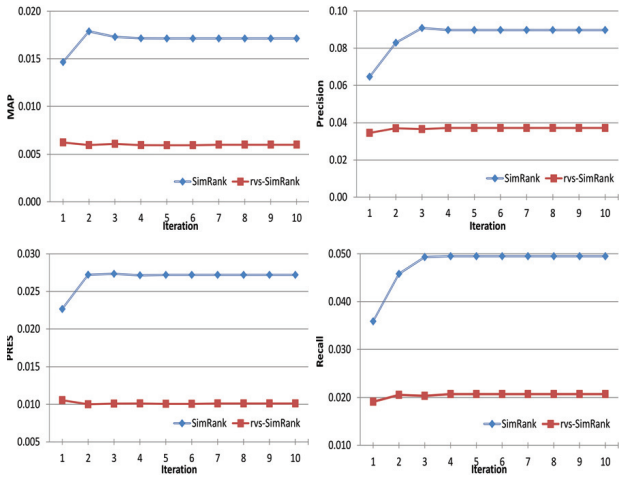

Figure 2: Accuracy of rvs-SimRank in comparison with SimRank.

Figure 3 shows the effectiveness of P-Rank in similarity computation in comparison with SimRank. P-Rank outperforms SimRank in terms of MAP, precision, PRES, and recall. The reason is that P-Rank solves the in-links consideration problem. To compute the similarity score between any pairs of papers, P-Rank not only considers the papers citing them but also considers those papers cited by them. In this way, P-Rank assigns a similarity score to some pairs of papers that are considered as dissimilar by SimRank such as (e, h) in Figure 1. Also, it assigns a similarity score to those pairs of papers that are considered as dissimilar by rvs-SimRank such as (b, c). P-Rank assigns a similarity score to 398,165,965 paper-pairs in ten iterations.

Figure 4 illustrates the effectiveness of PSimRank, C-Rank, and MatchSim in similarity computation in comparison with SimRank. On all iterations, PSimRank, C-Rank, and Match-

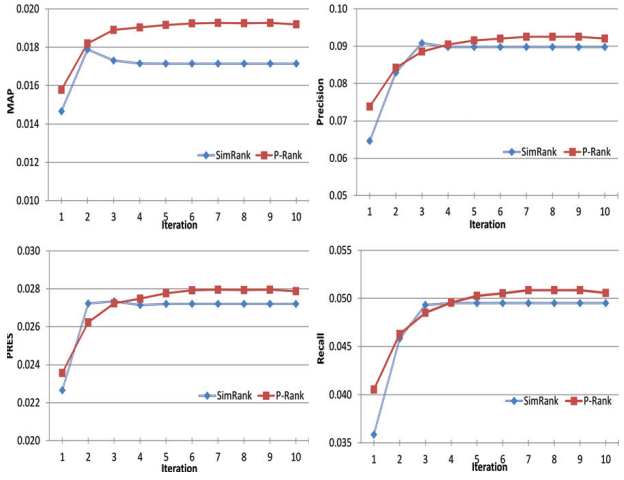

Figure 3: Accuracy of P-Rank in comparison with SimRank.

Sim *dramatically* outperform SimRank in terms of MAP, precision, PRES, and recall. The reason is that these measures solve the pairwise normalization problem. As shown in Table 1, the SimRank score of (b, c) is higher than that of (g, i); however, PSimRank, C-Rank, and MatchSim do not assign a higher similarity score to (b, c) than (g, i).

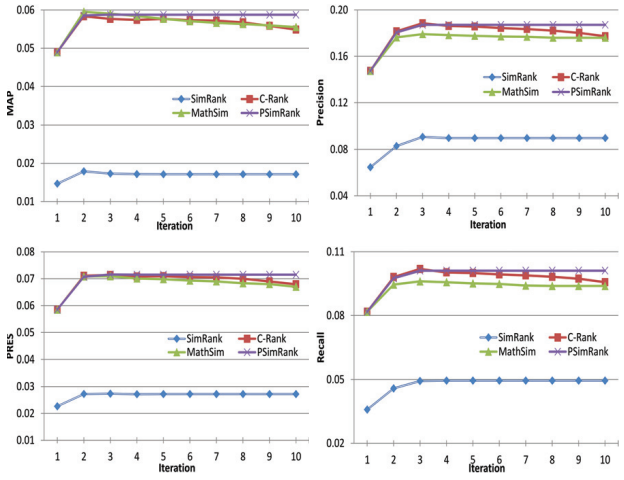

Figure 4: Accuracy of PSimRank, C-Rank, and MatchSim in comparison with SimRank.

PSimRank and C-Rank employ Jaccard Coefficient in similarity computation. On the first iteration, the similarity score of any paper-pairs (a, b) is computed *only* based on the Jaccard Coefficient score between I_a and I_b because there is not any paper-pair (i, j) between $I_a - I_b$ and I_b, and also between $I_b - I_a$ and I_a where $i = j$. In other words, scores obtained by the second and third parts of Equations (6) and (7) tend to be zero on the first iteration. On successive iterations i ($i \geq 2$), *only* the second and third parts of Equations (6) and (7) are computed iteratively by using similarity scores obtained on the previous iteration $i - 1$.

MatchSim utilizes maximum matching in similarity computation. On the first iteration, for any paper-pairs (a, b), the maximum matching between I_a and I_b is *identical* to their Jaccard Coefficient score. Therefore, MatchSim shows the same accuracy as PSimRank and C-Rank on the first iteration. PSimRank, C-Rank, and MatchSim assign a similarity score to 89,310,415 paper-pairs in ten iterations as SimRank.

Figure 5 shows the effectiveness of SimRank* and SimRank in similarity computation. On all iterations, SimRank* outperforms SimRank in terms of MAP, precision, PRES, and recall because SimRank* solves the level-wise computation problem. To compute similarity scores, SimRank* not only considers all the paths taken into account by SimRank but also considers those paths neglected by SimRank. SimRank* assigns a similarity score to those pairs of papers that are considered as dissimilar by SimRank since there are not paths with equal length from any common papers to both of them such as (e, h) in Figure 1. Furthermore, SimRank* is also able to compute the similarity scores based on RWR [16]. As an example, the SimRank score of paper-pair (b, g) in Figure 1 is zero; however, SimRank* regards papers b and g as likely similar as shown in Table 1. SimRank* assigns a similarity score to 170,411,955 paper-pairs in ten iterations.

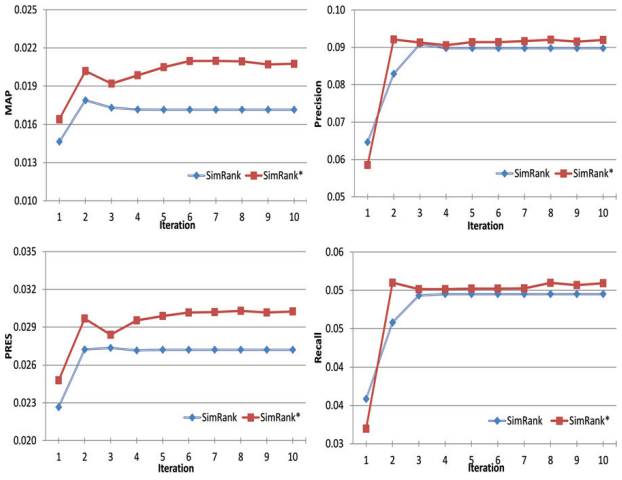

Figure 5: Accuracy of SimRank* in comparison with SimRank.

Now, we compare the effectiveness of SimRank and its variants based on their best shown accuracy in Figure 6. The best accuracy of SimRank, P-Rank, PSimRank, C-Rank, MatchSim, and SimRank* are observed on iteration 3, 7, 3, 3, 3, and 8, respectively. All measures outperform SimRank in terms of MAP, precision, PRES, and recall. It implies that the existing problems of SimRank *negatively* affect its effectiveness. Note that (1) still P-Rank has pairwise normalization and level-wise computation problems; PSimRank, C-Rank, and MatchSim have in-links consideration and level-wise computation problems; SimRank* has in-links consideration and pairwise normalization problems. (2) PSimRank, C-Rank, and MatchSim dramatically outperform P-Rank and SimRank* as well. These imply that the pairwise normalization problem has *more* negative effect on effectiveness of similarity computation than in-links consideration and level-wise computation problems.

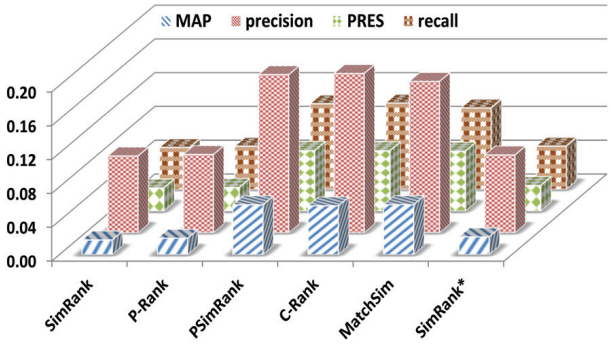

Figure 6: Accuracy of SimRank and its variants on their best accuracy.

5. CONCLUSION

SimRank is a well-known link-based similarity measure that has been applied in different applications as well as similarity computation for academic literature data. Despite of its current success, SimRank has the *three existing problems* as the in-links consideration, pairwise normalization, and level-wise computation. We discussed these problems extensively by providing a sample citation graph. In addition, we presented SimRank variants that have been provided solutions to SimRank problems; P-Rank solves the in-links consideration problem, PSimRank, C-Rank, and MatchSim solve the pairwise normalization problem, and SimRank* solves the level-wise computation problem. By conducting extensive experiments on a real-world dataset of academic papers, we showed that *all* SimRank variants (except rvs-SimRank) improve the accuracy of SimRank in similarity computation. Also, the results of our experiments implies that the pairwise normalization problem has *more* negative effect than two other problems on the effectiveness of SimRank.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (NRF-2014R1A2A1A10054151).

6. REFERENCES

[1] I. Antonellis, H. G. Molina, and C. C. Chang. Simrank++: Query Rewriting Through Link Analysis of the Click Graph. *PVLDB*, 1(1):408–421, 2008.

[2] Y. Cai, P. Li, H. Liu, J. He, and X. Du. S-SimRank: Combining Content and Link Information to Cluster Papers Effectively and Efficiently. In *Lecture Notes in Computer Science*, pages 317–329, 2008.

[3] D. Fogaras and B. Racz. Scaling Link-based Similarity Search. In *WWW*, pages 641–650, 2005.

[4] K. Fujita, Y. Kajikawa, J. Mori, and I. Sakata. Detecting Research Fronts Using Different Types of Weighted Citation Networks. *Engineering and Technology Management*, 32(1):129–146, 2014.

[5] Y. Fujiwara, M. Nakatsuji, H. Shiokawa, and M. Onizuka. Efficient Search Algorithm for SimRank. In *ICDE*, pages 589–600, 2013.

[6] J. Han, M. Kamber, and J. Pei. *Data Mining: Concepts and Techniques, Second Edition*. Morgan Kaufmann, San Francisco, 2006.

[7] G. He, H. Feng, C. Li, and H. Chen. Parallel SimRank Computation on Large Graphs with Iterative Aggregation. In *SIGKDD*, pages 543–552, 2010.

[8] J. Jeh and J. Widom. SimRank: A Measure of Structural-Context Similarity. In *SIGKDD*, pages 538–543, 2002.

[9] M. Kusumoto, T. Maehara, and K.-i. Kawarabayashi. Scalable Similarity Search for SimRank. In *SIGMOD*, pages 325–336, 2014.

[10] C. Li, J. Han, G. He, X. Jin, Y. Sun, Y. Yu, and T. Wu. Fast Computation of SimRank for Static and Dynamic Information Networks. In *EDBT*, pages 465–476, 2010.

[11] Z. Lin, M. R. Lyu, and I. King. MatchSim: A Novel Neighbor-based Similarity Measure with Maximum Neighborhood Matching. In *CIKM*, pages 1613–1616, 2009.

[12] W. Magdy and G. J. Jones. PRES: A Score Metric for Evaluating Recall-oriented Information Retrieval Applications. In *SIGIR*, pages 611–618, 2010.

[13] C. Manning, P. Raghavan, and H. Schutze. *Introduction to Information Retrieval*. Cambridge University Press, New York, 2008.

[14] M. Reyhani Hamedani, S. Kim, S. Lee, and D. Kim. On Exploiting Content and Citation Together to Compute Similarity of Scientific Papers. In *CIKM*, pages 1553–1556, 2013.

[15] Y. Saad. *Iterative Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2003.

[16] H. Tong, C. Faloutsos, and J.-Y. Pan. Fast Random Walk with Restart and Its Applications. In *ICDM*, pages 613–622, 2006.

[17] S. Yoon, J. Kim, J. Ha, S.-W. Kim, M. Ryu, and H.-J. Choi. Link-Based Similarity Measures Using Reachability Vectors. *The Scientific World Journal*, 2014, 2014.

[18] S. Yoon, S. Kim, and P. Sunju. C-Rank: A Link-based Similarity Measure for Scientific Literature Databases. *arXiv:1109.1059*, 2011.

[19] W. Yu, X. Lin, W. Zhang, L. Chang, and J. Pei. More is Simpler: Effectively and Efficiently Assessing Node-pair Similarities Based on Hyperlinks. *PVLDB*, 7(1):2150–8097, 2013.

[20] W. Yu and J. A. McCann. Efficient Partial-pairs Simrank Search on Large Networks. *PVLDB*, 8(5):569–580, 2015.

[21] W. Yu, W. Zhang, X. Lin, Q. Zhang, and J. Le. A Space and Time Efficient Algorithm for SimRank Computation. *World Wide Web*, 15(3):327–353, 2012.

[22] P. Zhao, H. Han, and S. Yizhou. P-Rank: a Comprehensive Structural Similarity Measure over Information Networks. In *CIKM*, pages 553–562, 2009.

[23] W. Zheng, L. Zou, Y. Feng, L. Chen, and D. Zhao. Efficient Simrank-based Similarity Join over Large Graphs. *PVLDB*, 6(7):493–504, 2013.