Gauging Correct Relative Rankings For Similarity Search

Weiren Yu†, Julie A. McCann†
†Imperial College London, United Kingdom
{weiren.yu, j.mccann}@imperial.ac.uk

ABSTRACT

One of the important tasks in link analysis is to quantify the similarity between two objects based on hyperlink structure. SimRank is an attractive similarity measure of this type. Existing work mainly focuses on absolute SimRank scores, and often harnesses an iterative paradigm to compute them. While these iterative scores converge to exact ones with the increasing number of iterations, it is still notoriously difficult to determine how well the relative orders of these iterative scores can be preserved for a given iteration. In this paper, we propose efficient ranking criteria that can secure correct relative orders of node-pairs with respect to SimRank scores when they are computed in an iterative fashion. Moreover, we show the superiority of our criteria in harvesting top-K SimRank scores and bucket orders from a full ranking list. Finally, viable empirical studies verify the usefulness of our techniques for SimRank top-K ranking and bucket ordering.

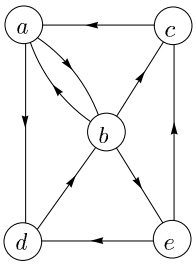
1. INTRODUCTION

The problem of identifying similar objects based on graph structure is a fundamental primitive in hyperlink analysis, arising in numerous applications, *e.g.,* anomaly detection, recommendation systems, and automated image annotation. It often demands a measure of closeness between two objects. For instance, Shortest distance can be regarded as a simple measure that counts only *one* path with minimum length to evaluate pair-wise similarity. Recently, SimRank has been proposed by Jeh and Widom [1] as a promising measure of affinity between two nodes. It follows the idea that "two nodes are similar if they are referenced by similar nodes". Due to its recursion, SimRank can count *multiple* paths with different lengths between two nodes to evaluate similarity, which is a substantial improvement over shortest distance.

To serve ranking purposes, this paper focuses on SimRank measure due to its two advantages:

1. Unlike PageRank that is query-independent, SimRank scores between all nodes and a given query q can yield a query-specific ranking list of all nodes imposed by q.
2. SimRank scores can rank both nodes and node-pairs, in contrast to PageRank only for node ranking.

CIKM'15, October 19-23, 2015, Melbourne, VIC, Australia.
 ISBN 978-1-4503-3794-6/15/10 ...$15.00.
DOI: http://dx.doi.org/10.1145/2806416.2806610.



Node-Pairs	Relative Ranking			
	$k=1$	$k=2$	$k=3$	$k=\infty$
(a,e)	1	1	1	1
(c,e)	1	1	2	2
(a,c)	3	4	4	4
(b,d)	3	4	5	5
(c,d)	3	3	3	3
(a,b)	–	6	6	6
(a,d)	–	6	7	7
(b,c)	–	6	8	8
(b,e)	–	9	9	9
(d,e)	–	–	10	10
Accuracy C^{k+1}	0.36	0.216	0.130	–

Figure 1: Relative Ranking *w.r.t.* Iteration k

While most existing work [1–7] focuses on iterative computations of *absolute* SimRank scores, the issue of gauging the correctness of their *relative* ranking has received little attention. To the best of our knowledge, the previous iterative methods to compute SimRank often first empirically set the total number of iterations, k, and then use the k-th iterative SimRank score $s_k(a,b)$ to estimate the exact solution $s(a,b)$. For accuracy guarantee, Lizorkin *et al.* [5] showed an upper bound of the gap between $s_k(a,b)$ and $s(a,b)$:

$$0 \leq s(a,b) - s_k(a,b) \leq C^{k+1}, \qquad \forall k, \quad \forall a,b \qquad (1)$$

where $0 < C < 1$ is a decay factor. However, from the ranking perspective, it seems hard to use this (*absolute*) gap to determine how well the relative ranking with respect to k-th iterative SimRank scores can be preserved, since even a large gap in Eq.(1) does not necessarily imply incorrect *relative* ranking of objects, as illustrated in Example 1.

Example 1. *Figure 1 depicts how the relative ranking of node-pairs in graph G (with respect to their k-th iterative SimRank scores sorted in descending order) is updated when the number of iteration k increases. The last column of the table ($k=\infty$) shows the "true" relative ranking with respect to exact SimRank scores. The (absolute) error bounds of SimRank for every iteration k are depicted in the last line.*

From the table, it can be noticed that the relative ranking tends to the "true" one as k increases. In fact, when $k=3$, the relative ranking with respect to $s_3(\star,\star)$ has become the same as the "true" one, but the (absolute) gap of SimRank scores between $s_3(\star,\star)$ and exact $s(\star,\star)$ is not suitably small, which can be bounded by $C^{k+1} = 0.6^{3+1} = 0.130$. [1] □

Example 1 indicates that the correctness of relative ranking of node-pairs may not be solely judged by the *absolute*

[1]As previously used in [1], we set the decay factor $C = 0.6$ for SimRank computation in Example 1.

gap of SimRank scores in Eq.(1). For ranking purposes, the correct relative orders of node-pairs are more important than their absolute SimRank scores. Thus, it is imperative to identify efficient ranking criteria that can guarantee the correct relative order of node-pairs with respect to their SimRank scores during the iterative computation.

Hence, we consider the following problem, referred to as *Relative Ranking Criterion of SimRank* (RRCS).

Given the number of iterations k, for every two node-pairs (a,b) and (c,d), our goal is to find δ_k such that

$$s_k(a,b) - s_k(c,d) \geq \delta_k \text{ implies } s(a,b) \geq s(c,d). \quad (2)$$

The main challenge in RRCS is the determination of δ_k, which can be used as a threshold to check whether the k-th iterative SimRank scores $s_k(\star,\star)$ of any two node-pairs are well separated. If affirmative, we can conclude at iteration k that their final ("true") relative ranking with respect to the exact SimRank scores $s(\star,\star)$ can be consistently preserved. Another direct benefit of RRCS lies in its high effectiveness for top-K ranking and bucket ranking.

Contributions. We make the following contributions:

1. A proposed ranking criterion to gauge the correct relative order of node-pairs with respect to their SimRank scores. (Section 3)
2. Two induced ranking criteria for SimRank top-K ranking and bucket ordering. (Section 4)
3. Viable empirical studies showing the effectiveness of these criteria for ranking objects. (Section 5)

We contend that our techniques for RRCS yield a promising systematic method, which is also applicable to many other metrics, such as PageRank, Random Walk with Restart, ObjectRank, and SimFusion.

Related Work. Ranking nodes or node-pairs based on link structure is an important application of SimRank similarity. Nonetheless, most existing work mainly concerns *absolute* SimRank scores computation [1, 3, 5, 6, 8–10]. As for *relative* ranking, there is only one work by Lizorkin *et al.* who made the first effort in Proposition 2 of [5] to establish the following ranking accuracy estimate:

$$s(a,b) - s(a,d) \geq C^{k+1} \text{ implies } s_k(a,b) \geq s_k(a,d).^2 \quad (3)$$

A striking difference between Eqs.(3) and (2) is the logical order — Eq.(3) infers k-th relative ranking from exact one, whereas Eq.(2) infers exact relative ranking from k-th iterative one. In fact, in an iterative process, exact $s(\star,\star)$ are unknown beforehand. Hence, we can only use k-th iterative information in $s_k(\star,\star)$ to infer exact $s(\star,\star)$. In this sense, Eq.(2) is more useful in practice.

There has been work on other accuracy estimates [3,5] for iterative SimRank computation. Lizorkin *et al.* [5] are the first to propose an (absolute) error estimate for SimRank:

$$0 \leq s(a,b) - s_k(a,b) \leq C^{k+1}, \qquad \forall k, \quad \forall a,b$$

Based on this bound, it is easy to find out the total number of iterations required to guarantee a given accuracy. However, from the ranking perspective, the relative order of node-pairs can be correctly preserved *before* C^{k+1} becomes small.

[2] [5] uses $R_k(\star,\star)$ to denote k-th iterative SimRank scores, in contrast to $s_k(\star,\star)$ in this paper.

Later, Zheng *et al.* [3] showed the gap between two consecutive SimRank iterations:

$$0 \leq s_{k+1}(a,b) - s_k(a,b) \leq C^{k+1}, \qquad \forall k, \quad \forall a,b$$

with the aim to deduce an upper bound for SimRank score of each node-pair. Although the accuracy estimates in [3,5] can be used as stopping criteria in iterative SimRank computation, they may not guarantee correct relative ranking.

Recently, SimRank top-K queries [7, 10] have witnessed growing interests. Lee *et al.* [7] proposed a novel random walk based method to identify top-K nearest neighbors with respect to a given query q based on SimRank scores $s(q,\star)$. If our ranking criteria were incorporated into their method, the speedup for top-K nodes would be more pronounced. Fujiwara *et al.* [10] leveraged a min-heap structure as well as a Cauchy-Schwarz inequality to prune unlikely nodes for top-K SimRank search. However, their approach is based on the matrix decomposition, which is different from ours.

Our RRCS criteria can also applied to iterative PageRank computation. Most existing convergence criteria for PageRank (*e.g.,* [11, 12]) are based on the *absolute* difference between 1) the k-th iterative PageRank values and the ideal ones, or 2) the two consecutive PageRank iterations. Several work has exploited geometric distance [13] and Kendall's τ distance [14] for top-K PageRank rankings. When our RRCS criteria are integrated into these methods, the relative rankings of PageRank can be efficiently obtained as well.

2. PRELIMINARIES

In this section, we briefly revisit the SimRank background. For presentation ease, we use its matrix representation [4].

Notations. The following notations are used in the paper.

$[\mathbf{X}]_{x,y}$	(x,y)-entry of matrix $\mathbf{X} \in \mathbb{R}^{n \times n}$
$\|\mathbf{X}\|_{\max}$	max-norm matrix of $\mathbf{X}$ $(= \max_{i,j=1}^{n} \|x_{i,j}\|)$
$\|\mathbf{X}\|_{\infty}$	∞-norm matrix of $\mathbf{X}$ $(= \max_{i=1}^{n} \sum_{j=1}^{n} \|x_{i,j}\|)$

Consider a graph $G = (V, E)$ with node set V and edge set E. Let $\mathbf{S}$ be the SimRank matrix whose entry $[\mathbf{S}]_{a,b}$ is the similarity $s(a,b)$ between nodes a and b, and let $\mathbf{Q}$ be the backward transition matrix whose entry $[\mathbf{Q}]_{a,b} = 1/(\text{in-degree of } a)$ if there is an edge $b \to a$, and 0 otherwise. Then, $\mathbf{S}$ satisfies the following recursion:

$$\mathbf{S} = \max\{C \cdot (\mathbf{Q} \cdot \mathbf{S} \cdot \mathbf{Q}^T), \mathbf{I}\}, \quad (4)$$

where $0 < C < 1$ is a decay factor, $(\star)^T$ is matrix transpose, and $\mathbf{I}$ is an identity matrix with compatible dimension, and $\max\{\star\}$ is an element-wise maximum operation.

Intuitively, $[\mathbf{S}]_{a,b}$ depends on two terms in Eq.(4):

1) The term $[\mathbf{Q} \cdot \mathbf{S} \cdot \mathbf{Q}^T]_{a,b}$ includes the average similarity of (a,b)'s in-neighbor pairs, which implies that "two nodes are similar if their in-neighbors are similar".

2) $\max\{\star, \mathbf{I}\}$ guarantees that the diagonals of $\mathbf{S}$ are all 1s, corresponding to "every node is maximally similar to itself".

Let $\mathbf{S}_k$ be the k-th iterative SimRank matrix. Then, the exact $\mathbf{S}$ in Eq.(4) can be iteratively computed as

$$\mathbf{S}_{k+1} = \max\{C \cdot (\mathbf{Q} \cdot \mathbf{S}_k \cdot \mathbf{Q}^T), \mathbf{I}\} \text{ with } \mathbf{S}_0 = \mathbf{I}. \quad (5)$$

3. RELATIVE RANKING OF SIMRANK

We provide a ranking criterion to gauge the correct relative order of node-pairs with respect to the SimRank scores. The main result in this section is as follows.

THEOREM 1. *For every two node-pairs (a,b) and (c,d), if*

$$[\mathbf{S}_k]_{c,d} - [\mathbf{S}_k]_{a,b} \geq \tfrac{C^k}{1-C^k} \cdot \max_{i \neq j}\{[\mathbf{S}_k]_{i,j}\}, \quad \forall k = 1, 2, \cdots$$

then it necessarily follows that $[\mathbf{S}]_{c,d} \geq [\mathbf{S}]_{a,b}$. □

(The proof will be given later after some discussion.)

Intuitively, Theorem 1 provides an efficient criterion for SimRank relative order preservation, by finding a suitable δ_k in Eq.(2), which is practically small and easy-to-compute. It suggests that, when SimRank is iteratively computed from Eq.(5), for any two node-pairs, if the gap of their k-th iterative scores is no less than $\delta_k := \frac{C^k}{1-C^k} \cdot \max_{x \neq y}\{[\mathbf{S}_k]_{x,y}\}$, then we can determine, at iteration k, their correct ("true") relative rankings with respect to the exact SimRank scores.

Other important applications of Theorem 1 are top-K ranking and bucket ordering, as will be seen in Section 4.

To prove Theorem 1, the following lemmas are needed.

LEMMA 1. *For every $k = 0, 1, \cdots$, and each $j = 0, 1, \cdots, k$,*

$$\|\mathbf{S}_k - \mathbf{S}_{k+j}\|_{\max} \leq C^j \cdot \|\mathbf{S}_{k-j} - \mathbf{S}_k\|_{\max}. \quad \square$$

PROOF. One can readily derive from Eq.(5) that

$$\mathbf{S}_k - \mathbf{S}_{k+j} = C \cdot \mathbf{Q} \cdot (\mathbf{S}_{k-1} - \mathbf{S}_{k+j-1}) \cdot \mathbf{Q}^T,$$

from which it follows inductively that

$$\mathbf{S}_k - \mathbf{S}_{k+j} = C^j \cdot \mathbf{Q}^j \cdot (\mathbf{S}_{k-j} - \mathbf{S}_k) \cdot (\mathbf{Q}^T)^j.$$

Take $\|\star\|_{\max}$ norm on both sides, and apply the fact that

$$\|[\mathbf{Q}]_{a,\star} \cdot \mathbf{X} \cdot [\mathbf{Q}^T]_{\star,b}\|_{\max} \leq \|\mathbf{X}\|_{\max} \text{ with } \mathbf{X} = \mathbf{S}_{k-j} - \mathbf{S}_k,$$

to the above equation, it follows that

$$\begin{aligned}\|\mathbf{S}_k - \mathbf{S}_{k+j}\|_{\max} &\leq C^j \cdot \|\mathbf{Q}^j \cdot (\mathbf{S}_{k-j} - \mathbf{S}_k) \cdot (\mathbf{Q}^T)^j\|_{\max} \\ &\leq C^j \cdot \|\mathbf{Q}^{j-1} \cdot (\mathbf{S}_{k-j} - \mathbf{S}_k) \cdot (\mathbf{Q}^T)^{j-1}\|_{\max} \\ &\leq \cdots \leq C^j \cdot \|\mathbf{S}_{k-j} - \mathbf{S}_k\|_{\max}. \quad \square\end{aligned}$$

Intuitively, for a current iteration k, Lemma 1 provides an accuracy estimate for predicting new SimRank in the future j iterations, by using old SimRank in the past j iterations.

LEMMA 2. *For $j = 1, 2, \cdots$, the following estimate holds:*

$$\|(\mathbf{I} - C^j \cdot (\mathbf{Q}^j \otimes \mathbf{Q}^j))^{-1}\|_\infty \leq \tfrac{1}{1-C^j}. \quad \square$$

PROOF. Since $\|\mathbf{Q}^j \otimes \mathbf{Q}^j\|_\infty \leq 1$, by using the fact that

$$(\mathbf{I} - \mathbf{X})^{-1} = \sum_{k=0}^{\infty} \mathbf{X}^k \text{ with } \mathbf{X} = C^j \cdot (\mathbf{Q}^j \otimes \mathbf{Q}^j),$$

we can obtain

$$\begin{aligned}\left\|\left(\mathbf{I} - C^j \cdot (\mathbf{Q}^j \otimes \mathbf{Q}^j)\right)^{-1}\right\|_\infty &= \left\|\sum_{k=0}^{\infty} C^{jk} (\mathbf{Q}^j \otimes \mathbf{Q}^j)^k\right\|_\infty \\ &\leq \sum_{k=0}^{\infty} C^{jk} = \tfrac{1}{1-C^j}. \quad \square\end{aligned}$$

Lemma 2 gives a neat bound for $\|(\mathbf{I} - C^j \cdot (\mathbf{Q}^j \otimes \mathbf{Q}^j))^{-1}\|_\infty$, which lays the foundation for the proof of Theorem 1. Such an upper bound is tight since it can be readily shown that "=" in Lemma 2 is attainable whenever every node in a graph has at least one incoming edge.

LEMMA 3. *For every $k = 1, 2, \cdots$, and each $j = 1, 2, \cdots, k$,*

$$\|\mathbf{S}_k - \mathbf{S}\|_{\max} \leq \tfrac{C^j}{1-C^j} \cdot \|\mathbf{S}_{k-j} - \mathbf{S}_k\|_{\max}. \quad \square$$

PROOF. We can readily verify by induction that

$$\mathbf{S}_{k+j} - \mathbf{S} = C^j \cdot \mathbf{Q}^j \cdot (\mathbf{S}_k - \mathbf{S}) \cdot (\mathbf{Q}^T)^j.$$

Thus, we have

$$\begin{aligned}\mathbf{S}_k - \mathbf{S}_{k+j} &= (\mathbf{S}_k - \mathbf{S}) - (\mathbf{S}_{k+j} - \mathbf{S}) \\ &= (\mathbf{S}_k - \mathbf{S}) - C^j \cdot \mathbf{Q}^j \cdot (\mathbf{S}_k - \mathbf{S}) \cdot (\mathbf{Q}^T)^j.\end{aligned}$$

Taking $vec(\star)$ operator on both sides, and then applying the tensor product $\otimes$ property, we have

$$vec(\mathbf{S}_k - \mathbf{S}_{k+j}) = (\mathbf{I} - C^j \cdot (\mathbf{Q}^j \otimes \mathbf{Q}^j)) \cdot vec(\mathbf{S}_k - \mathbf{S}).$$

On both sides, we first multiply by $(\mathbf{I} - C^j \cdot (\mathbf{Q}^j \otimes \mathbf{Q}^j))^{-1}$, and then take $\|\star\|_\infty$, which yields

$$\begin{aligned}\|\mathbf{S}_k - \mathbf{S}\|_{\max} &\leq \underbrace{\|(\mathbf{I} - C^j \cdot (\mathbf{Q}^j \otimes \mathbf{Q}^j))^{-1}\|_\infty}_{\text{by Lemma 2}} \cdot \underbrace{\|\mathbf{S}_k - \mathbf{S}_{k+j}\|_{\max}}_{\text{by Lemma 1}} \\ &\leq \tfrac{C^j}{1-C^j} \cdot \|\mathbf{S}_{k-j} - \mathbf{S}_k\|_{\max}. \quad \square\end{aligned}$$

Lemma 3 suggests that the accuracy of SimRank scores at iteration k can be estimated by utilizing the old SimRank in the past j iterations.

Combining Lemmas 1-3, we can prove Theorem 1.

PROOF OF THEOREM 1. Since the iterative SimRank score monotonically increases to the exact solution with respect to k, it follows that $[\mathbf{S}]_{x,y} \geq [\mathbf{S}_k]_{x,y}$, for every node-pair (x, y).

Setting $j = k$ in Lemma 3, we denote by

$$\delta_k = \tfrac{C^k}{1-C^k} \cdot \|\mathbf{I} - \mathbf{S}_k\|_{\max} = \tfrac{C^k}{1-C^k} \cdot \max_{x \neq y}\{[\mathbf{S}_k]_{x,y}\}.$$

Then, Lemma 3 can be rewritten as $\|\mathbf{S}_k - \mathbf{S}\|_{\max} \leq \delta_k$, which implies that, for every two node-pairs (a, b) and (c, d),

$$[\mathbf{S}]_{c,d} - [\mathbf{S}_k]_{c,d} \leq \delta_k \text{ and } [\mathbf{S}]_{a,b} - [\mathbf{S}_k]_{a,b} \geq 0.$$

Subtraction of the above two equations yields

$$[\mathbf{S}_k]_{a,b} - [\mathbf{S}_k]_{c,d} - \delta_k \leq [\mathbf{S}]_{a,b} - [\mathbf{S}]_{c,d}.$$

This implies that

$$\text{if } [\mathbf{S}_k]_{a,b} - [\mathbf{S}_k]_{c,d} - \delta_k \geq 0, \text{ then } [\mathbf{S}]_{a,b} - [\mathbf{S}]_{c,d} \geq 0,$$

which completes the proof. □

4. APPLICATIONS

To appreciate the utility of our relative ranking criteria for SimRank (RRCS), we next illustrate two real applications.

Notations. Let $\mathbf{T}$ be a permutation matrix that arranges all the entries of a vector in decreasing order, *i.e.,*

$$\hat{\mathbf{s}}_k := \mathbf{T} \cdot vec(\mathbf{S}_k) \text{ with } [\hat{\mathbf{s}}_k]_1 \geq [\hat{\mathbf{s}}_k]_2 \geq \cdots \geq [\hat{\mathbf{s}}_k]_{n^2},$$ [3]

Given the above $\mathbf{T}$, we also define $\hat{\mathbf{s}} := \mathbf{T} \cdot vec(\mathbf{S})$. Note that the entries in $\hat{\mathbf{s}}$ are generally not sorted in decreasing order, as opposed to those in $\hat{\mathbf{s}}_k$.

Top-K Ranking. One application of RRCS is to validate top-K node-pairs search, based on the following corollary.

COROLLARY 1. *For every iteration $p = 1, 2, \cdots$, if*

$$[\hat{\mathbf{s}}_p]_K - [\hat{\mathbf{s}}_p]_{K+1} \geq \tfrac{C^p}{1-C^p} \cdot \max_{x \neq y}\{[\mathbf{S}_p]_{x,y}\}, \tag{6}$$

then $[\hat{\mathbf{s}}]_1, \cdots, [\hat{\mathbf{s}}]_K$ are the top-K highest SimRank scores.

[3]Recall that $[\hat{\mathbf{s}}_k]_i$ denotes the i-th element of vector $\hat{\mathbf{s}}_k$, and $vec(\mathbf{S}_k)$ stacks columns of $\mathbf{S}_k$ on top of one another [9].

PROOF. As $[\hat{\mathbf{s}}_p]_1 \geq \cdots \geq [\hat{\mathbf{s}}_p]_K \geq \cdots \geq [\hat{\mathbf{s}}_p]_{n^2}$, it follows from Eq.(6) that, for all $i = 1, \cdots, K$, and $j = K+1, \cdots, n^2$,

$$[\hat{\mathbf{s}}_p]_i - [\hat{\mathbf{s}}_p]_j \geq \frac{C^p}{1-C^p} \cdot \max_{x \neq y}\{[\mathbf{S}_p]_{x,y}\}.$$

By Theorem 1, we have $[\hat{\mathbf{s}}]_i \geq [\hat{\mathbf{s}}]_j$, for all $i = 1, \cdots, K$, and $j = K+1, \cdots, n^2$, which implies that $[\hat{\mathbf{s}}]_1, \cdots, [\hat{\mathbf{s}}]_K$ are the top-K highest SimRank scores. □

Corollary 1 tells that the "true" top-K node-pairs *w.r.t.* the exact SimRank scores are the same as the top-K node-pairs *w.r.t.* the p-th iterative SimRank scores if the p-th iterative SimRank ranking scores between positions K and $K+1$ are well separated above a threshold.

Bucket Ordering. Another application is bucket ordering. In this case, we need to assign SimRank scores of $n \times n$ node-pairs to several "bucket" intervals, as shown in Corollary 2.

COROLLARY 2. *Let* $\delta_p := \frac{C^p}{1-C^p} \cdot \max_{x \neq y}\{[\mathbf{S}_p]_{x,y}\}$. *For every iteration* $p = 1, 2, \cdots$, *if for* $u, v = 1, 2, \cdots$,

$$[\hat{\mathbf{s}}_p]_K - [\hat{\mathbf{s}}_p]_{K+u} \geq \delta_p, \quad [\hat{\mathbf{s}}_p]_{K+u} - [\hat{\mathbf{s}}_p]_{K+u+v} \geq \delta_p,$$

then $[\hat{\mathbf{s}}]_{K+u}$ *is ranked between* $(K+1)$ *and* $(K+u+v-1)$.

PROOF. Analogous to the proof of Corollary 1, $[\hat{\mathbf{s}}_p]_K - [\hat{\mathbf{s}}_p]_{K+u} \geq \delta_p$ implies that $[\hat{\mathbf{s}}]_i \geq [\hat{\mathbf{s}}]_{K+u}$, for all $i = 1, \cdots, K$. Thus, the rank position of $[\hat{\mathbf{s}}]_{K+u}$ is after K.

Besides, from $[\hat{\mathbf{s}}_p]_{K+u} - [\hat{\mathbf{s}}_p]_{K+u+v} \geq \delta_p$ and $[\hat{\mathbf{s}}_p]_{K+u+v} \geq \cdots \geq [\hat{\mathbf{s}}_p]_{n^2}$ follows that $[\hat{\mathbf{s}}_p]_{K+u} - [\hat{\mathbf{s}}_p]_j \geq \delta_p$ for all $j = K+u+v, \cdots, n^2$. This implies that

$$[\hat{\mathbf{s}}]_{K+u} \geq [\hat{\mathbf{s}}]_j \text{ for all } j = K+u+v, \cdots, n^2.$$

Hence, the rank position of $[\hat{\mathbf{s}}]_{K+u}$ is before $K+u+v$.

Taking these together, we can obtain that $[\hat{\mathbf{s}}]_{K+u}$ is ranked between $(K+1)$ and $(K+u+v-1)$. □

Corollary 2 assigns SimRank score $[\hat{\mathbf{s}}_p]_{K+u}$ to a "bucket" that represents a rank interval $[K+1, K+u+v-1]$. Indeed, the top-K ranking in Corollary 1 is a special case of bucket ranking with two (interval) "buckets": $[1, K]$ and $[K+1, n^2]$.

5. EXPERIMENTS

We present an empirical study on real networks to evaluate the usefulness of our criteria for ranking node-pairs.

Datasets. Two real networks are used: 1) ENRON, an email communication network from Enron, with 367,662 edges and 36,692 nodes. 2) AMAZON, an Amazon product co-purchasing graph, with 1,234,877 edges and 262,111 nodes.

We implement iterative SimRank algorithm [5] [4] in Visual C++, and use a machine with an Intel Core(TM) 3.10GHz CPU and 8GB RAM, running Windows 7.

Results. By tying our ranking criterion of Corollary 2 to algorithm [5], Figure 2 shows how many distinct ranks can be identified (*i.e.,* the number of buckets) and how many elements per bucket (*i.e.,* bucket sizes) at the last iteration ($k = 10$)[5]. To ensure better visibility for top-ranked buckets, we omit the rightmost bucket (whose size is the largest, yet contains node-pairs with the smallest SimRank scores). The detailed information is depicted in Table 1, where we see that on ENRON, 8.6% of the smallest node-pairs cannot be ranked, whereas on AMAZON, this number increases to 58.2%. Among the top-100 ranking results, 91 (*resp.* 100) node-pairs on ENRON (*resp.* AMAZON) are exactly ranked. These indicate the effectiveness of our ranking criterion for identifying SimRanks of the top-ranked node-pairs.

[4]Our ranking criteria can also be applied to other iterative SimRank algorithms [1, 6, 15], yielding the same results.

[5]As used in [5], the total iteration number k is set to 5–10.

Dataset	% of pairs in exact ranking	# of pairs in exact top-100	% of pairs in last bucket
ENRON	41.7%	91	8.6%
AMAZON	34.5%	100	58.2%

Table 1: Statistical Information of Bucket Ranking

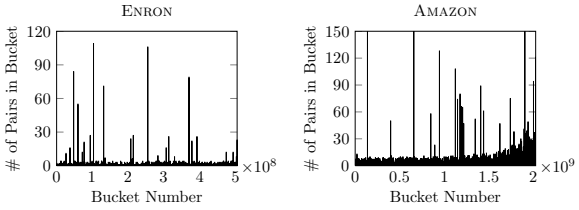


Figure 2: Bucket Ranking on Real Networks ($k = 10$)

6. CONCLUSIONS

This paper provides several useful relative ranking criteria for SimRank iterations. Important applications of our ranking criteria include top-K ranking and bucket ordering.

As a future avenue, we will incorporate these criteria for K nearest neighbor search and other similarity models [16, 17].

Acknowledgements. This research is supported by NEC Smart Water Network research project.

7. REFERENCES

[1] G. Jeh and J. Widom, "SimRank: a measure of structural-context similarity," in *KDD*, 2002.
[2] M. Kusumoto, T. Maehara, and K. Kawarabayashi, "Scalable similarity search for SimRank," in *SIGMOD*, 2014.
[3] W. Zheng, L. Zou, Y. Feng, L. Chen, and D. Zhao, "Efficient SimRank-based similarity join over large graphs," *PVLDB*, vol. 6, no. 7, pp. 493–504, 2013.
[4] W. Yu and J. A. McCann, "Efficient partial-pairs SimRank search for large networks," *PVLDB*, vol. 8, no. 5, pp. 569–580, 2015.
[5] D. Lizorkin, P. Velikhov, M. N. Grinev, and D. Turdakov, "Accuracy estimate and optimization techniques for SimRank computation," *VLDB J.*, vol. 19, no. 1, pp. 45–66, 2010.
[6] W. Yu, X. Lin, W. Zhang, and J. A. McCann, "Fast all-pairs SimRank assessment on large graphs and bipartite domains," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 7, pp. 1810–1823, 2015.
[7] P. Lee, L. V. S. Lakshmanan, and J. X. Yu, "On top-k structural similarity search," in *ICDE*, 2012.
[8] G. He, H. Feng, C. Li, and H. Chen, "Parallel SimRank computation on large graphs with iterative aggregation," in *KDD*, 2010.
[9] 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*, 2010.
[10] Y. Fujiwara, M. Nakatsuji, H. Shiokawa, and M. Onizuka, "Efficient search algorithm for SimRank," in *ICDE*, pp. 589–600, 2013.
[11] P. Berkhin, "Survey: a survey on PageRank computing," *Internet Mathematics*, vol. 2, pp. 73–120, 2005.
[12] R. S. Wills and I. C. F. Ipsen, "Ordinal ranking for Google's PageRank," *SIAM J. Matrix Analysis Applications*, vol. 30, no. 4, pp. 1677–1696, 2008.
[13] A. Borodin, G. O. Roberts, J. S. Rosenthal, and P. Tsaparas, "Link analysis ranking: algorithms, theory, and experiments," *ACM Trans. Internet Techn.*, vol. 5, no. 1, pp. 231–297, 2005.
[14] R. Fagin, R. Kumar, and D. Sivakumar, "Comparing top k lists," in *SODA*, pp. 28–36, 2003.
[15] W. Yu, X. Lin, and W. Zhang, "Fast incremental SimRank on link-evolving graphs," in *ICDE*, pp. 304–315, 2014.
[16] W. Yu and X. Lin, "IRWR: Incremental random walk with restart," in *SIGIR*, pp. 1017–1020, 2013.
[17] W. Yu, X. Lin, W. Zhang, Y. Zhang, and J. Le, "SimFusion+: Extending SimFusion towards efficient estimation on large and dynamic networks," in *SIGIR*, pp. 365–374, 2012.