SimRank: A Measure of Structural-Context Similarity

Glen Jeh and Jennifer Widom
1. Co-Founder at FriendDash
2. Stanford, Computer Science, department chair
SIGKDD 2002, Citation: 506 (Google Scholar)
Outline

• 1. Introduction
• 2. Related Work
• 3. Basic Graph Model
• 4. SimRank
• 5. Random Surfer-Pairs Model
• 6. Experimental Results
• 7. Conclusion
• Many applications require a measure of “similarity” between objects.
• One obvious example is the “find-similar-document” query, on traditional text corpora or the World-Wide Web [2].

Introduction (2/6)

• A similarity measure can be used to cluster objects, such as for collaborative filtering in a recommender system [7, 11, 20].


The authors propose the SimRank to calculate the similarity between two objects based on the \textit{structural context}.

\[
\begin{array}{cc}
  a & b \\
  \text{related} & \text{related} \\
  c & d \\
  \text{similar} & \text{similar}
\end{array}
\]
Figure 1: A small Web graph $G$ and simplified node-pairs graph $G^2$. SimRank scores using parameter $C = 0.8$ are shown for nodes in $G^2$. 
Figure 1: A small Web graph $G$ and simplified node-pairs graph $G^2$. SimRank scores using parameter $C = 0.8$ are shown for nodes in $G^2$. 
The authors run an iterative fixed-point algorithm on $G^2$ to compute what the authors call SimRank scores for the node-pairs in $G^2$. Scores can be thought of as “flowing” from a node to its neighbors. Each iteration propagates scores one step forward along the direction of the edges, until the system stabilizes.
Related Work (1/4)

• **Structural context** has been used and analyzed in specific applications, such as
  – bibliometrics
  – database schema-matching
  – hypertext classification.

• Most noteworthy from this field are the methods of co-citation [21] and bibliographic coupling [9].


Related Work (2/4)

• In the co-citation scheme

• In bibliographic coupling
Related Work (3/4)

• Co-citation scores between any two nodes are computed only from their immediate neighbors.
• SimRank can use the entire graph structure to determine the similarity between any two nodes.
Related Work (4/4)

• The intuitive **underlying model** for the similarity measure is based on “*random surfers*”.
  – A concept which is also used in [16] to provide an intuitive model for the *PageRank algorithm*.
  – The authors formalize and extend the model using *expected-f distances*.

Basic Graph Model (1/2)

• In web pages or scientific papers (homogeneous domains):
  – nodes represent documents
  – a directed edge \(<p, q>\) from \(p\) to \(q\) corresponds to a reference (hyperlink or citation) from document \(p\) to document \(q\).
Basic Graph Model (2/2)

• In a user-item domain (bipartite):
  – represent both users and items by nodes in $V$.
  – A directed edge $<p, q>$ corresponds to a purchase (or other expression of preference) of item $q$ by person $p$.

• For a node $v$ in a graph, denote by $I(v)$ and $O(v)$ the set of in-neighbors and out-neighbors of $v$, respectively.
Basic SimRank Equation (1/7)

• If $a = b$ then $s(a, b)$ is 1. Otherwise,

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

(1)

• $C$ is a constant between 0 and 1.
• If $I(a) = \text{empty set}$ or $I(b) = \text{empty set}$, then $s(a, b) = 0$. 
Basic SimRank Equation (2/7)

• If \( a = b \) then \( s(a, b) \) is 1. Otherwise,

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

(1)

• Divide by the total number of in-neighbor pairs, \( |l(a)| \times |l(b)| \), to normalize.
• The similarity between \( a \) and \( b \) is the average similarity between in-neighbors of \( a \) and in-neighbors of \( b \).
Basic SimRank Equation (3/7)

- If the graph size is $n$, there are $n^2$ SimRank equations.
  - SimRank scores are symmetric, i.e., $s(a, b) = s(b, a)$. 

17
Basic SimRank Equation (4/7)

• Similarity propagates in $G^2$ from node to node, with the sources of similarity being the singleton nodes.
The cycles in $G^2$, caused by the presence of cycles in $G$, allow similarity to flow in cycles

- such as from {Univ, ProfB} back to {ProfA, ProfB} in the example.

Similarity scores are thus *mutually reinforced*.
Basic SimRank Equation (6/7)

• Consider the constant $C$, which can be thought of either as a confidence level or a decay factor.

• The similarity of $x$ with itself is 1, but we probably don’t want to conclude that
  $s(c, d) = s(x, x) = 1$. 
Basic SimRank Equation (7/7)

• Let $s(c, d) = C \cdot s(x, x)$
  
  – meaning that we are less confident about the similarity between $c$ and $d$ than we are between $x$ and itself.
Figure 2: Shopping graph $G$ and a simplified version of the derived node-pairs graph $G^2$. Bipartite SimRank scores are shown for $G^2$ using $C_1 = C_2 = 0.8$. 

Bipartite SimRank (1/3)
Bipartite SimRank (2/3)

• Let \( s(A, B) \) denote the similarity between persons \( A \) and \( B \). \((A \neq B)\)

\[
s(A, B) = \frac{C_1}{|O(A)||O(B)|} \sum_{i=1}^{O(A)} \sum_{j=1}^{O(B)} s(O_i(A), O_j(B)) \tag{2}
\]

– Neglecting \( C_1 \), equation (2) says that the similarity between persons \( A \) and \( B \) is the average similarity between the items they purchased.
Bipartite SimRank (3/3)

• Let $s(c, d)$ denote the similarity between items $c$ and $d$. ($c \neq d$)

\[
s(c, d) = \frac{C_2}{|I(c)||I(d)|} \sum_{i=1}^{I(c)} \sum_{j=1}^{I(d)} s(I_i(c), I_j(d))
\]  

– Neglecting $C_2$, equation (3) says that the similarity between items $c$ and $d$ is the average similarity between the people who purchased them.
The Minimax Variation (1/4)

- “Recursive notion of structural-context similarity”.
- Goal:
- Finding similarity between undergraduate students and between courses based on the students’ history of courses taken.
The Minimax Variation (2/4)

• For example:

![Diagram showing relationships between Computer Science, Sociology-related electives, Computer Science Required courses, and English-related electives]

• *It may be meaningless to compare A’s CS courses with B’s electives.*
The Minimax Variation (3/4)

• To compare each of B’s courses c with only the one course taken by A which is most similar to c.

• Define the intermediate terms \( s_A(A, B) \) and \( s_B(A, B) \) (for \( A \neq B \)) as follows:

\[
\begin{align*}
  s_A(A, B) &= \frac{C_1}{|O(A)|} \sum_{i=1}^{O(A)} \max_{j=1}^{O(B)} s(O_i(A), O_j(B)) \\
  s_B(A, B) &= \frac{C_1}{|O(B)|} \sum_{j=1}^{O(B)} \max_{i=1}^{O(A)} s(O_i(A), O_j(B))
\end{align*}
\]
The Minimax Variation (4/4)

\[ s(A, B) = \min(s_A(A, B), s_B(A, B)) \]  \hspace{1cm} (4)

to be the similarity between students A and B.
Computing SimRank Naïve Method (1/3)

• Let $n$ be the number of nodes in $G$.
  
  – For each iteration $k$, keep $n^2$ entries $R_k(\ast, \ast)$ of length $n^2$, where $R_k(a, b)$ gives the score between $a$ and $b$ on iteration $k$.
  
  – Start with $R_0(\ast, \ast)$ where each $R_0(a, b)$ is a lower bound on the actual SimRank score $s(a, b)$:

$$R_0(a, b) = \begin{cases} 
0 & \text{(if } a \neq b) \\
1 & \text{(if } a = b) 
\end{cases}$$
Computing SimRank Naïve Method (2/3)

• To compute $R_{k+1}(a, b)$ from $R_k(\ast, \ast)$, use equation (1) to get:

$$R_{k+1}(a, b) = \frac{C}{|I(a)||I(b)|} \sum_{i=1}^{|I(a)|} \sum_{j=1}^{|I(b)|} R_k(I_i(a), I_j(b))$$  \hspace{1cm} (5)

$R_{k+1}(a, b) = 1$ for $a = b$.

• On each iteration $k + 1$, update the similarity of $(a, b)$
  – by using the similarity scores of the neighbors of $(a, b)$ from the previous iteration $k$ according to equation (1).
Computing SimRank Naïve Method (3/3)

• Time and space requirements:
• Space required is simply $O(n^2)$ to store the results $R_k$.
• Let $d_2$ be the average of $|l(a)| \times |l(b)|$ over all node-pairs $(a, b)$.
• The time required is $O(Kn^2d_2)$. 
Computing SimRank Pruning

• Consider only node-pairs within a radius of $r$ from each other.
• There are on average $d_r$ such neighbors for a node, then there will be $n \times d_r$ node-pairs.
• Let $d_2$ is the average of $|l(a)| \times |l(b)|$ for pages $a, b$ close enough to each other.
  – The time complexity: $O(Knd_r d_2)$.
  – The space complexity: $O(nd_r)$. 
A brief Summary

• 1. SimRank
• 2. SimRank is applied to the Bi-partite graph
• 3. The Minmax Variation
• 4. The Naïve compute for SimRank
• 5. The Pruning
Limited-Information Problem (1/4)

Figure 3: Little information is available for $A$, which is cited only by $B$. 
Figure 3: Little information is available for $A$, which is cited only by $B$.

$A_m$ is shown as a better match for $A$ than $A_1$, since $A_m$’s other citer is $B'$, which is similar to $B$. 
Limited-Information Problem (3/4)

• On the other, the authors don’t want to eliminate unpopular documents from consideration or popular documents to be favored for every query.
  
  – If eliminated the constant factor $1/|I(b)|$ from equation (1), then documents $b$ with a very high popularity would have a high similarity score with any other document $a$.

$$s(a, b) = \frac{C}{|I(a)||I(b)|} \sum_{i=1}^{\|I(a)\|} \sum_{j=1}^{\|I(b)\|} s(I_i(a), I_j(b))$$  \hspace{1cm} (1)
Limited-Information Problem (4/4)

• Weigh the final results of the algorithm by popularity, using the asymmetric formula

\[ s_P(a, b) = s(a, b) \cdot |I(b)|^P \]  \hspace{1cm} (6)

• where the constant \( P \) belongs to \((0, 1)\) is a parameter adjustable by the end user.
Random Surfer-Pairs Model

• It is important to have an intuition for the similarity scores produced by the algorithm.
• For this the authors provide an intuitive model based on “random surfers”.
Random Surfer-Pairs Model **Expected Distance**

- Let $u, v$ be any two nodes in $H$, a strongly connected graph.
- Define the expected distance from $u$ to $v$ as

\[ d(u, v) = \sum_{t: u \sim \tau v} P[t]l(t) \]  

- The summation is taken over all **tours** $t$ (**paths** that may have cycles) which start at $u$ and end at $v$. 
Random Surfer-Pairs Model Expected Distance

• For a tour $t = \langle w_1, \ldots, w_k \rangle$, the length $l(t)$ of $t$ is $k-1$, the number of edges in $t$.

• The probability $P[t]$ of traveling $t$ is
  \[
  \prod_{i=1}^{k-1} \frac{1}{|O(w_i)|}
  \]
  or 1 if $l(t) = 0$.

• The expected distance is exactly the expected number of steps a random surfer.

\[
\begin{align*}
  \frac{1}{|O(w_1)|} & \quad w_1 \\
  & \quad \quad \quad \quad w_2 \\
  & \quad \quad \quad \quad \quad \quad \ldots \\
  & \quad \quad \quad \quad \quad \quad \quad \quad w_k
\end{align*}
\]
Random Surfer-Pairs Model: Expected Meeting Distance

- The expected meeting distance \( m(a, b) \) between \( a \) and \( b \) is the expected number of steps required before two surfers meet.
Random Surfer-Pairs Model Expected Meeting Distance

• The expected meeting distance \( m(a, b) \) between \( a \) and \( b \) is the expected number of steps required before two surfers.

Figure 4: Sample graph structures.
Random Surfer-Pairs Model Expected Meeting Distance

• The expected meeting distance $m(a, b)$ between $a$ and $b$ is the expected number of steps required before two surfers.

Figure 4: Sample graph structures.
Random Surfer-Pairs Model Expected Meeting Distance

- The expected meeting distance $m(a, b)$ between $a$ and $b$ is the expected number of steps required before two surfers meet.

Figure 4: Sample graph structures.
Random Surfer-Pairs Model Expected Meeting Distance

- Each node $(a, b)$ of $V^2$ can be thought of as the present state of a pair of surfers in $V$.

- An edge from $(a, b)$ to $(c, d)$ in $G^2$ says that in the original graph $G$:
  - one surfer can move from $a$ to $c$ while the other moves from $b$ to $d$.

- A tour in $G^2$ of length $n$ represents a pair of tours in $G$ also having length $n$. 
Random Surfer-Pairs Model Expected Meeting Distance

• The EMD \( m(a, b) \) is the expected distance in \( G^2 \) from \( (a, b) \) to any singleton node \((x, x)\),

\[
m(a, b) = \sum_{t: (a,b) \leadsto (x,x)} P[t]l(t) \tag{8}
\]

– The sum is taken over all tours \( t \) starting from \((a, b)\) which touch a singleton node at the end and only at the end.

– However, this definition would cause problems in defining distances for nodes from which some tours lead to singleton nodes while others lead to \((a, b)\).
Random Surfer-Pairs Model **Expected-\(f\) Meeting Distance**

• **Map all distances to a finite interval:**
  – Compute the expected \(f(l(t))\), for a nonnegative, monotonic function \(f\) which is bounded on the domain \([0,1)\). “The expected \(f\)-meeting distance.”

• \(f(z) = c^z\), where \(c\) belongs to \((0, 1)\) is a constant.
  – Closer nodes have a lower score (meeting distances of 0 go to 1 and distances of infinite go to 0), matching our intuition of similarity.
Random Surfer-Pairs Model Expected-$f$ Meeting Distance

• Define $s'(a, b)$, the similarity between $a$ and $b$ in $G$ based on expected-$f$ meeting distance, as

$$s'(a, b) = \sum_{t : (a, b) \leadsto (x, x)} P[t] c^l(t)$$

(9)

![Diagram](a)

![Diagram](b)

![Diagram](c)
Random Surfer-Pairs Model Equivalence to SimRank (1/6)

• Suppose a surfer is at $u$ belongs to $V$.
  – At the next time step, he chooses one of $O_1(u),\ldots,O_{|O(u)|}(u)$, each with probability $1/|O(u)|$.
• Upon choosing $O_i(u)$, the expected number of steps he will still have to travel is $d(O_i(u), v)$.

\[
d(u, v) = 1 + \frac{1}{|O(u)|} \sum_{i=1}^{|O(u)|} d(O_i(u), v)
\]
Random Surfer-Pairs Model Equivalence to SimRank (2/6)

- If \( a = b \) then \( s'(a, b) = s(a, b) = 1 \).
- If there is no path in \( G^2 \) from \((a, b)\) to any singleton nodes, in which case \( s'(a, b) = 0, s(a, b) = 0 \) in equation (5) as well.

\[
R_{k+1}(a, b) = \frac{C}{|I(a)||I(b)|} \sum_{i=1}^{\left|I(a)\right|} \sum_{j=1}^{\left|I(b)\right|} R_k(I_i(a), I_j(b)) \quad (5)
\]

- Otherwise, consider the tours \( t \) from \((a, b)\) to a singleton node in which the first step is to the out-neighbor \( O_z((a, b)) \).
Random Surfer-Pairs Model Equivalence to SimRank (3/6)

- For each $t'$ may derive a corresponding $t$ by appending the edge $<(a, b), O_z((a, b))>$ at the beginning.
  - Let $T$ be the bijection that takes each $t'$ to the corresponding $t$.
  - If the length of $t'$ is $l$, then the length of $t = T(t')$ is $l+1$. 

\[ (a, b) \rightarrow O_z(a,b) \rightarrow \ldots \rightarrow (x,x) \]
Random Surfer-Pairs Model Equivalence to SimRank (4/6)

• The probability of traveling $t$ is

$$P[t] = \frac{1}{|O((a,b))|} P[t'] = \frac{1}{|O(a)||O(b)|} P[t']$$

• There is a one-to-one correspondence between such $t$ and tours $t'$ from $O_z((a, b))$ to a singleton node.
Random Surfer-Pairs Model Equivalence to SimRank (5/6)

- Now split the sum in (9) according to the first step of the tour $t$ to write:

\[
s'(a, b) = \sum_{z=1}^{\mid O((a,b)) \mid} \sum_{t': O_z((a,b)) \sim (x,x)} P[T(t')] c^{l(T(t'))} \\
= \sum_{z=1}^{\mid O((a,b)) \mid} \sum_{t': O_z((a,b)) \sim (x,x)} \frac{1}{\mid O(a) \mid \mid O(b) \mid} P[t'] c^{l(t')} + 1 \\
= \frac{c}{\mid O(a) \mid \mid O(b) \mid} \sum_{z=1}^{\mid O((a,b)) \mid} \sum_{t': O_z((a,b)) \sim (x,x)} P[t'] c^{l(t)} \\
= \frac{c}{\mid O(a) \mid \mid O(b) \mid} \sum_{i=1}^{\mid O(a) \mid} \sum_{j=1}^{\mid O(b) \mid} s'(O_i(a), O_j(b)) \quad (10)
\]
Theorem. *The SimRank score, with parameter \(C\), between two nodes is their expected-\(f\) meeting distance traveling back-edges, for \(f(z)=C^z\).

- Thus, two nodes with a high SimRank score can be thought of as being “close” to a common “source” of similarity.
A brief Summary

• 1. Random Surfer
• 2. The expected distance => the expected meeting distance
• 3. The EMD in $G$ => the EMD in $G^2$.
• 4. Solve the infinite problem($\text{EMD} \Rightarrow \text{EfMD}$).
• 5. The $\text{EfMD} \Rightarrow \text{SimRank}$
Experimental Results

• Experiments on two data sets.
  – Scientific research papers from ResearchIndex (http://www.researchindex.com)
    • 688,898 cross-references among 278,628 papers
  – The transcripts of 1030 undergraduate students in the School of Engineering at Stanford University.
    • Each transcript lists all the courses that the student has taken so far in his undergraduate career, an average of about 40 courses per student.
Experimental Results

• Consider objects $p$, generating lists of objects similar to $p$.

• The procedure for evaluating is as follows:
  – 1. Generate a set $\text{top}_{A,N}(p)$ of the top $N$ objects most similar to $p$, according to algorithm A.
  – 2. For each $q$ belongs to $\text{top}_{A,N}(p)$, compute $\sigma(p, q)$, where $\sigma$ is a coarse domain-specific similarity measure.
    • Return the average $\sigma_{A,N}(p)$ of these scores.
Experimental Results

• The number $\sigma_{A,N}(p)$: the average “actual” similarity to $p$ of the top $N$ objects that algorithm $A$ decides are similar to $p$.
  – Restricted the experiments to those objects $p$ for which co-citation had at least 50 candidates to consider, or $|c(p)| \geq 50$. 
Experimental Results

• $\sigma_{R,N}(p)$ : the average of $\sigma(p, q)$ for $N$ objects $q$ randomly chosen from $c(p)$.

• Measure the performance of algorithm A on object $p$ using the difference :

  \[- \delta_{A,N}(p) = \sigma_{A,N}(p) - \sigma_{R,N}(p) \]

  \[- \Delta_{A,N} : The \ average \ of \ \delta_{A,N}(p) \ over \ all \ p, \ the \ final \ score \ for \ algorithm \ A. \]
Experimental Results

Scientific Paper

- $\sigma^C(p, q) = \text{fraction of } q\text{'s citations also cited by } p\text{.}$
- $\sigma^T(p, q) = \text{fraction of words in } q\text{'s title also in } p\text{'s title.}$
- The bipartite variant of the SimRank algorithm:

$$s_2(a, b) = \frac{C_2}{|I(a)||I(b)|} \sum_{i=1}^{|I(a)|} \sum_{j=1}^{|I(b)|} s_1(I_i(a), I_j(b))$$

- $C_1 = C_2 = 0.8.$
Figure 5: SimRank and co-citation on scientific papers.
Figure 6: SimRank and co-citation on scientific papers for varying N.
Experimental Results Students and Courses

• The course-similarity metric is based on departments:

• $\sigma_D(p, q) = 1$ if $p$ and $q$ are courses from the same department, and $\sigma_D(p, q) = 0$ otherwise.
Minimax variation, $C_1 = C_2 = 0.8$.

$$s_A(A, B) = \frac{C_1 |O(A)|}{|O(B)|} \sum_{i=1}^{O(A)} \max_{j=1}^{O(B)} s(O_i(A), O_j(B))$$

$$s_B(A, B) = \frac{C_1 |O(B)|}{|O(A)|} \sum_{j=1}^{O(B)} \max_{i=1}^{O(A)} s(O_i(A), O_j(B))$$

$$s(A, B) = \min(s_A(A, B), s_B(A, B))$$

Co-citation scores, which are very poor
0.161 for $N = 5$
0.147 for $N = 10$

Figure 7: SimRank on courses for increasing iterations.
Conclusion

• Formalize the recursive notion of structural-context similarity and defined SimRank scores.
  – Presented variations of SimRank that are applicable to different domains.
  – Presented a fixed-point algorithm for computing SimRank scores, as well as methods to reduce its time and space requirements.
Conclusion

• Defined a “random-surfer” model by which to interpret solutions to the SimRank equations.
  – The model is based on the concept of expected meeting distance (EMD).

• Experiments Results, showing significant improvement over simpler co-citation measures.
Comment SimRank++

• The drawback of SimRank:
Comment SimRank++

- The drawback of SimRank:
Comment SimRank++

• The drawback of SimRank: