Unified and Incremental SimRank: Index-free Approximation with Scheduled Principle

Fanwei Zhu zhufanwei@zju.edu.cn
Yuan Fang yfang@smu.edu.sg
Kai Zhang drogozhang@gmail.com
Kevin Chen-Chuan Chang kcchang@illinois.edu
Hongtai Cao hongtai2@illinois.edu
Zhen Jiang jzjzjzzju@zju.edu.cn
Minghui Wu mhwu@zucc.edu.cn

Problem

Graphs are ubiquitous nowadays, requiring effective similarity measures based on their link structures.

- SimRank is a popular link-based similarity measure on graphs which enables a variety of applications with different modes of querying.

SimRank queries: S(Q)

SimRank Problems		Query Q=(A,B)	Output
General definition	Partial-pair	$A \subseteq V$ $B \subseteq V$	$s(u,v)$: a A-by-B similarity matrix, with each entry $[S]_{u,v} = s(u,v)$
Popular modes	Single-pair	A={u} B={v}	$s(u,v)$: a single SimRank similarity score between u and v
	Single-source	A={u} B=V	$[S]_u$: a $\|V\|$-by-1 similarity vector, with each entry $[S]_{u,v} = s(u,v)$
	All-pair	A=V B=V	$[S]$: a $\|V\|$-by-$\|V\|$ similarity matrix, with each entry $[S]_{u,v} = s(u,v)$

Motivations:

a) Distinct modes of SimRank: it is desirable to support all different modes in a unified manner by one algorithm

b) Specific accuracy requirement: it is desirable to support flexible tradeoffs of efficiency and accuracy.

c) Dynamic graphs with frequent updates: it is desirable to support efficient online computation without relying indexes.

Goal: fast approximation for all modes of SimRank queries

Proposed approach

UISIM: a unified and incrementally-enhanced framework to efficiently process different modes of SimRank queries.

Unification of computation space

All tours $T_Q = P_A \bowtie P_B$ aggregate to the exact scores, while a subset of tours gives an approximation.

- Organize T_Q into disjoint subsets $T_Q = T_Q^1 \cup \ldots \cup T_Q^\eta$ with T_Q^i is more important than T_Q^{i-1}
- Incrementally approximate S(Q) through iterations with iteration-i computing an SimRank increment over T_Q^i

Benefit-based prioritized approx.

Partition P_A and P_B based on their importance, and schedule the assemblies based on their “benefit” to the overall approximation.

- Partition partial tours by their hub length $L_h(p)$ (i.e.,number of hubs they pass through)

$$P_u = P_u^0 \cup \ldots P_u^\eta \ s.t. \ \forall p \in P_u^i, L_h(p) = i$$

- Schedule assembly $P_u^i \bowtie P_u^j$ earlier than $P_u^{i'} \bowtie P_u^{j'}$ if:

$$i + j \leq i' + j' \& |i - j| \leq |i' - j'|$$

Index-free sharing of computation

Efficiently realize the scheduled approximation without relying on any precomputed indexes.

- Extend tours in P_u^{i-1} with hub-length-0 extension tours to obtain partial tours in P_u^i

$$r^i(u|v) = \sum_{h \in H_u^{i-1}} r^{i-1}(u|h) \cdot r^0(h|v)$$

- Skip “mis-matching” partial tours when generating full tours

$$R(P_u^i \bowtie P_v^j) = \sum_{x \in X} \sum_{l \leq M} \frac{1}{C^l} (r^{i,l}(u|x) \cdot r^{i,l}(v|x))$$

Experimental evaluation

- **Datasets**:

Dataset	Directed	Nodes	Edges	Purpose
4Area	no	12 413	91 192	Parameter study, and comparison to baselines
WikiVote	yes	1 300	39 456	
CondMat	no	23 133	93 497	
enwiki2013	yes	4 206 785	101 355 853	Comparison to baselines
it2014	yes	41 291 594	1 150 725 436	
Friendster	no	65 608 366	1 806 067 135	
Gnutella	yes	62 586	147 892	Scalability study
Dblp	no	2 073 13	25 759 41	

- **Baselines**: the single- pair solution BLPMC; the single-source solutions ProbeSim, PRSim and SimPush ; the all-pair solution FLP, and the SimRank Join solution TreeWand.

- **Experiental results (partial):**

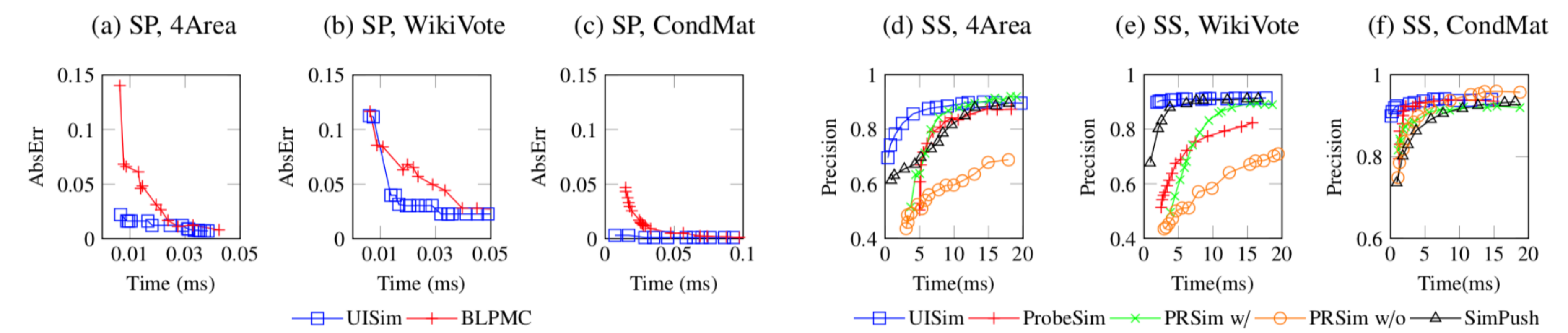

Fig 1. Comparison of accuracy against time with baselines in single pair and single source modes.

- **Conclusion**: 1) UISim always needs less time to achieve the same accuracy as its baseline; 2) UISim achieves a good accuracy very fast while the baselines do not perform well within limited time.

TKDE poster @ 38th IEEE International Conference on Data Engineering (ICDE), May 9–12, 2022, Kuala Lumpur, Malaysia