Semsim++ : Boosting Simrank with Semantic

Chen Liguo
School of Data Science
Fudan University
Shanghai, China
17307110182@fudan.edu.cn

Qi Yifan
School of Data Science
Fudan University
Shanghai, China
17307110182@fudan.edu.cn

The problem of measuring "similarity" of objects arises in many applications, especially estimating the similarity of a pair of nodes in an information network draws extensive interest in numerous fields like social network, recommender system and knowledge graph. A popular and well-studied similarity measure is SimRank, which compares the similarity of a pair of nodes based on the similarity of their neighbors. SimRank's popularity stems from its simple, declarative definition and its efficient scalable computation. However, even it has been used widely, there are still some inaccuracy has been observed in the similarity query. And most importantly, SimRank only considers the structure of network and ignores the semantic information, for example, the labels or weight of nodes and edges. A natural question that we can ask can SimRank be enriched with semantic while preserving its advantages?

A variant method of SimRank is SemSim, which allows to take semantic information into consideration. The probabilistic framework that we develop for SemSim is anchored in a careful modification of SimRank's underlying random surfer model, and SemSim model allows to inject different definition of the semantic similarity measuring. It also employs Importance Sampling and pruning techniques based on the unique properties of SemSim.

While considering some applications in the natural world, we design some modifications of SemSim model, which can enhance the performance on some specific local pattern which appears frequently in real world information networks. And we also propose some techniques to improve the space and time efficiency. And all the modification of modular which means that it can be adjusted and replaced according to different tasks, which not only allows us to adjust the influence of semantic and structure to the query answer but also can adjust or replenish the pattern we use according to specific situation. Our experiments demonstrate the robustness of our model comparing to the exist SemSim model.

Keywords—similarity, semantic, SimRank, SemSim

I. Introduction

Estimating node similarity in information networks is the cornerstone of many applications, e.g., retrieving similar users in social networks, and a fundamental component in numerous network analysis algorithms, such as link prediction and clustering.

In this work we consider SimRank, a well-studied similarity measure for information networks. The intuition behind SimRank is that similar objects are referenced by similar objects, and thus it quantifies node similarity based on the compound similarity of their neighbors. SimRank's popularity based on its efficient computation. However, despite its wide adaptation, it has been observed that for many applications SimRank may yield inaccurate estimations [1][2], as it focuses solely on the network structure and ignores the semantic information conveyed in the node/edge labels.

SemSim model enriches SimRank with semantics while preserving its intuitive and efficient computation. Besides the structural similarity, SemSim also considers semantic similarity and edge weights, yielding an effective and comprehensive measure.

We demonstrate the problem that we tackle with an illustrative example.

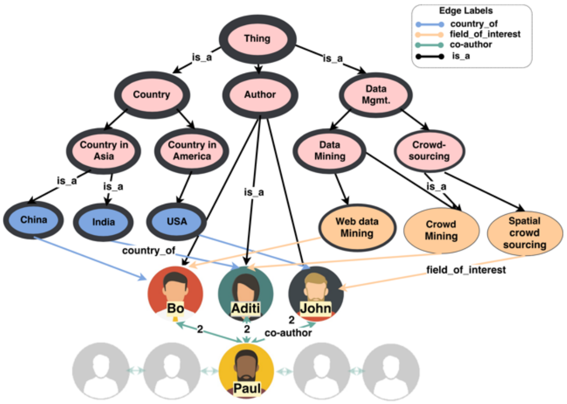

Figure 1 Example of information network

The simple information network depicted in Figure 1 represents a bibliographic database. It includes nodes describing authors, countries and research fields, with edges linking authors to their co-authors, country of origin and fields of interest. A semantic taxonomy is also reflected in this network where entities are linked to their hypernyms, as indicated by the "is-a" edges. Edge weights reflect the strength of the relations. To visually represent the prevalence of a concept in the dataset, we use the width of the borders surrounding the nodes.

We wish to determine which of the authors, Bo or John, is more similar to Aditi. In a traditional SimRank system, based on the structure of this network, Bo is more similar to Aditi than John because both of Bo and Aditi comes from an Asian country.

However, the previous method ignored a fact that the common filed shared by John and Aditi is Crowdsourcing, which is relative minority and more particular than Data Mining and should have a greater effect on similarity. And their origin countries are all highly prevalent compared to the authors' fields of interest, thus the latter is more informative and should have greater effect on similarity. Consequently, John is more similar to Aditi than Bo, even though they reside in different continents.

But there is still some inaccuracy need to fix, and some of them is caused by the inconsistency of the information network and others is that the semantic information is nor fully exploited. We can lower the error rate through taking some special pattern into consideration.

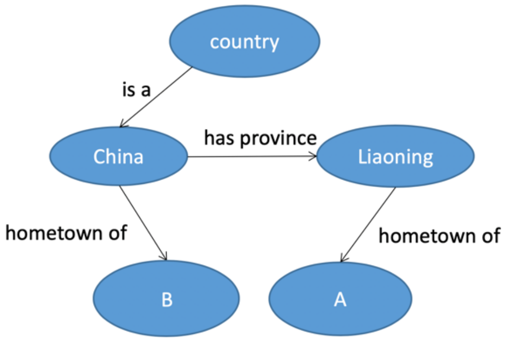

Figure 2 Same objects in different layers

A typical pattern is shown in figure 2, which is a common inconsistency in information network. This pattern we call it Malposition where the objects of same class such as A and B are in different layer. A's hometown is Liaoning which is a province of China while B's hometown is China. This inherent structural inconsistency can trace back to the data gathering and database designing stage. But it is difficult to fix in a given dataset and we want to lower the misleading influence in SimRank. So, we design a technique which will take a leap in the random surfer process to ensure the random surfer process will meet at the same layer.

In section II we will introduce related work that we refer to and in section three we will propose our SemSim++ framework. In section IV we will provide some experiment results to support our conclusion and in Section V we give a conclusion and some future work directions.

II. Related Work

SimRank is a similarity measuring method based on an intuition that the similarity of two nodes based on the similarity of their neighbors. SimRank received a widely adaptation and application for its simple and declarative definition and its efficient, scalable computation. However, it has been observed that for many applications, SimRank may yield inaccurate similarity estimations, due to the fact that it focuses on the network structure and ignores the semantics conveyed in the node/edge labels.

Several Refinements for SimRank have been proposed in the literature. For instance, SimRank++ [3] is a variant of SimRank that also consider edge weights but ignored semantic. And most importantly, scalability is not addressed. Other works only consider semantic in meta-paths which is composed of edges with specific label. But it is not always sufficed to accurately differentiate objects. Alternatively, several semantic measures have been proposed. But these methods ignored network structure and only considered ontological and Information Content (IC). In an attempt to fully account for both structure and semantic, recent work abandon SimRank and rely instead on representation learning, using techniques such as node embedding. This approach often outperforms a naïve combination of structural and semantic similarity measures, a key drawback is the result is hard to explain and interpret, as is often the case of machine learning.

The SemSim [4] model takes some adaptation to the traditional SimRank model to enrich it while preserving its advantages. It refines SimRank with by weighting nodes' neighbor similarity with their semantic similarity and edge weights. The definition of semantic similarity measure is modular and can replaced in different situations, as long as it satisfied three intuitive conditions that are typically satisfied by existing measures. And it is proved that a faster convergence is guaranteed.

Consider the fact that exact computation is expensive in large graphs even with speed-up techniques with graph reduction, SemSim model

III. Our Work

A. *Preliminaries:*

We first explain the data model we used in our setting, then proposed the SemSim++ method.

1) *Graph Model:*

Following, we refer to the objects graph as a Heterogeneous Information Network (HIN) [5], a flexible graph model that can capture and integrate various types of data. Let V be the domain of vertices, L the domain of labels.

Definition 3.1 (Heterogeneous Information Network). A HIN is a directed weighted graph $G = (V, E, \Phi, \Psi, W)$, where: $V \subseteq \mathcal{V}$ is a finite set of nodes, $E \subseteq V \times V$ is a set of edges, $\Phi: V \to \mathcal{L}$ and $\psi: E \to \mathcal{L}$ are node and edge labeling functions, resp., and $W: E \to \mathbb{R}^+$ is an edge weight function.

Edges in HIN have weights, for example, the *co-author* edge in Figure 1 has weights representing the number of their co-authored papers. For a node v, we denote the set of its in and out neighbors as $I(v)$ and $O(v)$.

In many cases, the HIN is composed of two subgraphs. The first one contains individual nodes and their relations, and the second one is an ontological graph, showing the taxonomy of concepts. The ontological graph will be used in computing semantical similarity. When semantic information is not included, one can enrich the graph by aligning it with publicly available ontology.

2) *Similarity notion:*

A semantic-rich graph model contains additional knowledge that is captured by the label and weight of edges and nodes. So, we start with the regular SimRank model similarity notion. Next we refined it to SemSim model and considering all information.

As we describe above, the similarity between two objects is determined by the similarity of their neighbors. So intuitively we can give the SimRank similarity score recurrently. If $u = v$ then $simrank(u, v) = 1$, else SimRank value is given by the following recursive formula:

$$sim(u, v) = \frac{c}{N_{u,v}} \sum_{i}^{|I(u)|} \sum_{j}^{|I(v)|} sim(I_i(u), I_j(v))$$

Where c is a decay factor between (0,1). $N_{u,v} = |I(u)| \cdot |I(v)|$ is the product of degree of given nodes and $sim(\cdot,\cdot)$ is the SimRank score of neighboring pair nodes. If $I(u)$ or $I(v)$ are empty sets we define it as 0.

Considering the effect of weight of edges, which means the strength of connection between two objects, a modification of SimRank recursive formula is defined as follow:

$$sim(u,v) = \frac{c}{N_{u,v}} \sum_{i}^{|I(u)|} \sum_{j}^{|I(v)|} W(I_i(u),u) \cdot W(I_j(v),v) \cdot sim(I_i(u), I_j(v))$$

Where $W(I_i(u),u)$ means the weight between $I_i(u)$ and u. For example, in Figure 1, the weight of edge between Bo and Paul is 2 means that Bo and Paul co-authored 2 papers in the past.

3) Semantic-aware Similarity:

Now we inject a semantic portion into the formula:

$$Sem(u,v) = \frac{sem(u,v) \cdot c}{N_{u,v}} \sum_{i}^{|I_i(u)|} \sum_{j}^{|I_j(v)|} sim\left(I_i\,(u), I_j(v)\right) \cdot W(I_i(u),u) \cdot W\left(I_j(v),v\right)$$

According to the formula above we can see that the intuition of Semantic-aware similarity is unchanged and therefore, the similarity of neighboring pairs of nodes u, v is proportional to the semantic similarity of their neighbors as well.

4) Semantic Similarity:

Multiple semantic similarity measures have been proposed in literature. In general, any similarity function $sem(\cdot,\cdot)$ can be employed in SemSim, as long as it satisfies the following constraints. For all $u, v \in V$:

1) Symmetry. $sem(u,v) = sem(v,u)$
2) Maximum self similarity. $sem(u,u) = 1$
3) Fixed value range. $sem(u,v) \in (0,1]$

We next briefly overview a simple and effective semantic measure that we have used in our experiments. Lin [6] is an Information Content (IC)-based measure that is defined over concept taxonomies. The IC of a node quantify as the negative of its log likelihood:

$$IC(v) = -\log\,(P[v])$$

Where $P(v)$ denotes the frequency of v. Put it simple, the more prevalent a concept is, the lower its IC value.

And given two nodes u and v, their Lin score is defined as:

$$Lin(u,v) = \frac{2 \cdot IC\big(LCA(u,v)\big)}{IC(u) + IC(v)}$$

Where $LCA(u,v)$ is the lowest common ancestor of u and v in the taxonomy.

Considering the fact that only if IC value in (0,1] the Lin formula is satisfied, and as the frequency of a node increase, the difference barely makes any sense on the degree of similarity, then we make a modification of the IC formula:

$$sigmoidIC(u) = -\log\,(\frac{sigmoid(freq)}{sigmoid(N)})$$

$$sigmoid(freq) = \frac{1}{1 + e^{-freq}}$$

It is worth to note that even the log change in previous IC formula lower the influence in large number, we think it still need reconsideration. And the $sigmoid$ change helps us better control the effect in large frequency instance.

B. Naïve iteration model:

We can use the former recursive formula to compute the SemSim iteratively. The similarity of a pair in k+1 step of iteration is computed from k-th step.

$$R_0(u,v) = \begin{cases} 0, u \neq v \\ 1, u = v \end{cases}$$

$$R_{k+1}(u,v) = \frac{sem(u,v) \cdot c}{N_{u,v}} \sum_{i}^{|I(u)|} \sum_{j}^{|I(v)|} R_k(I_i(u), I_j(v)) \cdot W(I_i(u),u) \cdot W(I_j(v),v)$$

But in fact the iteration method is not applicable in large-scale problem as the time complexity is endurably high.

C. Random Suffer-pairs Model:

Inspired by the random walk model of SimRank, the similarity of two nodes can be computed as follow efficiently. The key idea of random walk is that two suffers start at given object nodes and then randomly walk on the graph backward, and then compute the similarity by the average meeting time before they meet. In this section, we will firstly introduce Semantic-Aware Random Walks (SARW) and then make some improvement.

1) Semantic-aware Random Walks:

In order to make the explanation more clearly, we use the definition of a node-pair graph G^2, in which each node represents an ordered pair of nodes from G.

Edge $e = ((u,u'),(v,v')) \in G^2$ if and only if both (u,v) and $(u',v') \in$ G. And we define the weights of G^2 as:

$$W_{G^2} = W(u,v) \cdot W(u',v')$$

Using G^2, we can simply do SARW just considering one start instead of two. We call a node $(u,v) \in V^2$ is a singleton node if $u = v$, so that arriving at a singleton node in G^2 equals to a meet in original G.

In order to incorporate semantics and weights, SemSim devises the following distribution:

Definition 3.1 (Semantic-Aware Probability Distribution). The probability a random surfer traveling G^2 in a current node (u,u') would next move to its out-neighbor (v,v') is:

$$P[(u,u') \rightarrow (v,v')] = \frac{W\big((u,u'),(v,v')\big) \cdot sem(v,v')}{\sum_{i=1}^{|O((u,u'))|} W\big((u,u'),O_i(u,u')\big) \cdot sem(O_i(u,u'))}$$

Due to the definition of G^2, a random walk in G^2 represents a pair of walks in G. If we define a walk like $w = (w_1, \dots, w_k)$, the probability $P[w]$ of traveling in this path is:

$$P[w] = \prod_{i=1}^{k-1} P[w_i \rightarrow w_{i+1}]$$

2) Adaptation of SARW:

The structure of an information network or a knowledge graph is variable and complicated. To deal with the structural diversity, we need to add some exploration into SARW.

For example, Figure 2 is a part of an information network with information about *person A* and *person B*. The hometown of *person B* is *China*, a country, however, the hometown of *person A* is *Liaoning*, a province of China. The distance from *B* to *country* is 2, but the *has province* relation adds an extra hop in the path from *A* to *country*. This pattern is very common in a large graph, but SARW cannot find the similarity between person A and per will cause a failure to meet. As a result, important evidence of A and B's similarity is ignored.

To improve SARW's ability to find similarities in similar situations, we adapt SARW with an exploration process. In the process of random walk, every time we need to move we have a small chance of moving two steps. If two random surfers *a* and *b* start from *A* and *B* respectively, the extra one step gives *a* a chance to meet *b* in the *country* node.

To conclude, with a small probability we move two steps a time. Each move is based on the distribution defined in 3.4.1. This small probability of exploration can be represented by adding extra edges and nodes into G^2. The weights of additional edges can be defined as:

$$W_{G^2} = \begin{cases} W(u,v)\cdot W(u',v'), common\ situation \\ \alpha W(u,v)\cdot W(u',n)\cdot W(n,v'), exploration \end{cases}$$

α is the small probability of exploration. The G2 of Figure 2 is shown in Figure 3:

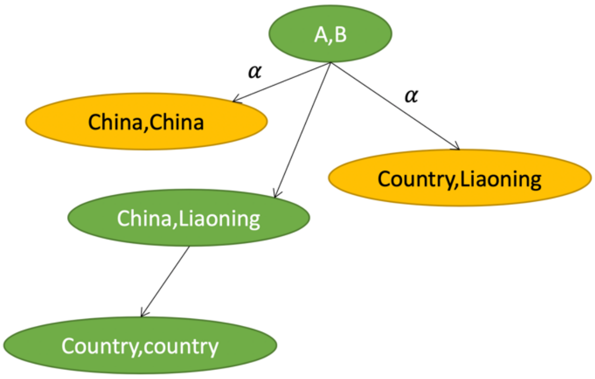

Figure 3 local G^2 graph for hometown

D. MC Framework:

It has been shown in literature that the SimRank score equals to an exponential expected distance $E[c^{\tau}]$. We can apply a Monte Carlo (MC) approximation framework by sampling separated random walks and estimated the similarity score using the average meeting distance:

$$\frac{1}{n_w}\sum_{l=1}^{n_w} c^{\tau_l}$$

, where τ_l denotes the number of steps before two suffers meet.

To sample from a self-defined distribution, we need to use Importance Sampling (IS). Importance Sampling is a general technique for estimating properties of distribution while only having samples generated from a different one.

$$E_P\left[c^{l(w)}\right] = \sum \frac{P(w)\cdot Q(w)\cdot c^{l(w)}}{Q(w)} \approx \frac{1}{n_w}\sum_{i=1}^{n_w} c^{l(w_i)}\frac{P(w_i)}{Q(w_i)}$$

, where Q is a distribution that we can sample from. As a result, we get an unbiased estimator of $c^{l(w)}$ under the distribution P using samples drawn from Q.

In our work:

$$\mathrm{sim(u,v) = sem(u,v)}\cdot E_P\left[c^{l(w)}\right] = sem(u,v)\cdot E_Q\left[\frac{P(w)\cdot c^{l(w)}}{Q(w)}\right]$$

In real world implementation, we can choose the standard uniform distribution to simplify.

IV. Support

Considering the example that we proposed in figure 1. In this information network, we wish to determine which of the authors, Bo or John, is more similar to Aditi. Applying our model to the given information network in figure 1 we find out that we can successfully capture the semantic information in the graph.

According repeated experiments results, the average similarity between Aditi and Bo is 0.3905, and the similarity of Aditi and John is 0.4494, which means that John is more similar to Aditi than Bo. Considering the fact that the common field shared by Aditi and Bo is relatively minority and should have a larger effect on the similarity, the results matches the real situation.

The experiments show that our model is able to capture the semantic factor when evaluating similarity between two nodes in given information network.

V. Conclusion

In this paper we present SemSim++ which is based on the SemSim model which measures the similarity of two objects in an Information Network. SemSim++ preserves the intuitive definition and scalable computation and contains some adaptation considering some specific pattern in the Information Network to adjust the effect of semantic reasonably. It is an optimization of exist SemSim model and we think it works better. And the random walk framework employs Importance Sampling along with an effective pruning technique and maintains a negligible error rate.

Several interesting directions are left for future research. First, in practice, information networks are often dynamic and may induce uncertainty, hence it would be important to extent SemSim++ to such settings. The use of parallelism [7] and compact indexing mechanisms [8] to achieve further speedup are also an interesting direction for future work. Last, we have focused here only on single-pair queries. We intend on developing optimizations facilitating single-source and top-k similarity queries. [9]

VI. Reference

[1] [1] Yizhou Sun, Jiawei Han, Xifeng Yan, Philip S Yu, and Tianyi Wu. 2011. Path- sim: Meta path-based top-k similarity search in heterogeneous information networks. PVLDB (2011).

[2] [1] Wikipedia [n. d.]. Wikipedia: SimRank. https://en.wikipedia.org/wiki/SimRank. ([n. d.]).

[3] [1] Ioannis Antonellis, Hector Garcia Molina, and Chi Chao Chang. 2008. Simrank++: query rewriting through link analysis of the click graph. Proc. VLDB Endow. 1, 1 (August 2008), 408–421

[4] [1] Tova Milo, Amit Somech, and Brit Youngmann. Boosting SimRank with Semantics. 12.

[5] [1] Chuan Shi, Yitong Li, Jiawei Zhang, Yizhou Sun, and S Yu Philip. 2017. A survey of heterogeneous information network analysis. TKDE (2017).

[6] [1] DekangLin.1998.Aninformation-theoreticdefinitionofsimilarity..InICML.

[7] [1] YuanzheCai, GaoCong, XuJia, HongyanLiu, JunHe, JiahengLu, and Xiaoyong Du. 2009. Efficient algorithm for computing link-based similarity in real world networks. In ICDM'09. IEEE.

[8] [1] ZhenguoLi, YixiangFang, QinLiu, JiefengCheng, ReynoldCheng, and John Lui. 2015. Walking in the cloud: Parallel simrank at scale. PVLDB (2015).

[9] [1] Pei Lee, Laks VS Lakshmanan, and Jeffrey Xu Yu. 2012. On top-k structural similarity search. In ICDE.