软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2014,25(11):2602−2615 [doi: 10.13328/j.cnki.jos.004578]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

信息网络中一个有效的基于链接的结点相似度度量*

张应龙[1,2], 李翠平[1], 陈 红[1]

[1](数据工程与知识工程教育部重点实验室(中国人民大学),北京 100872)

[2](华东交通大学 软件学院,江西 南昌 330045)

通讯作者: 李翠平, E-mail: cuiping_li@263.net

摘 要: 信息网络无处不在.通过把网络中的对象抽象为点,把对象之间的关系刻画为边,相应的信息网络就可以用图来表示.图中结点相似度计算是图数据管理中的基本问题,在很多领域都有运用,比如社会网络分析、信息检索和推荐系统等.其中,著名的相似度度量是以 Personalized PageRank 和 SimRank 为代表.这两种度量本质都是以图中的路径来定义,然而它们侧重的路径截然不同.为此,提出了一个度量 SuperSimRank.它不仅涵盖了这些路径,而且考虑了 Personalized PageRank 和 SimRank 两者都没有考虑的路径,从而能够更加体现出这种链接关系的本质.在此基础上对 SuperSimRank 进行了理论分析,从而提出了相应的优化算法,使得计算性能从最坏情况 $O(kn^4)$提高到 $O(knl)$.这里,k 是迭代次数,n 是结点数,l 是边数.最后,通过实验验证了 SuperSimRank 优于 SimRank 和 Personalized PageRank,同时验证了优化算法在各种情况下都是有效的.

关键词: 随机游走;相似度度量;SimRank;Personalized PageRank

中图法分类号: TP311

中文引用格式: 张应龙,李翠平,陈红.信息网络中一个有效的基于链接的结点相似度度量.软件学报,2014,25(11):2602−2615. http://www.jos.org.cn/1000-9825/4578.htm

英文引用格式: Zhang YL, Li CP, Chen H. Effective link-based measure of node similarity on information networks. Ruan Jian Xue Bao/Journal of Software, 2014,25(11):2602−2615 (in Chinese). http://www.jos.org.cn/1000-9825/4578.htm

Effective Link-Based Measure of Node Similarity on Information Networks

ZHANG Ying-Long[1,2], LI Cui-Ping[1], CHEN Hong[1]

[1](Key Laboratory of Data Engineering and Knowledge Engineering of Ministry of Education of China (Renmin University of China), Beijing 100872, China)

[2](School of Software, East China Jiaotong University, Nanchang 330045, China)

Corresponding author: LI Cui-Ping, E-mail: cuiping_li@263.net

Abstract: Information networks are ubiquitous. These networks can be modeled by graphs where nodes refer to objects and edges are relationships between objects in the networks. Measuring nodes similarity in a graph is a fundamental problem of graph data management. There are many real-world applications based on similarity between objects, such as networks analyses, information retrieval and recommendation systems. A number of link-based similarity measures have been proposed. Both SimRank and personalized PageRank are the most popular and influential similarity measures. The two measures differ from each other because they depend on different paths. This paper proposes a similarity measure that takes advantages of both SimRank and personalized PageRank. The new measure strengthens the principle of SimRank: "Two objects are similar if they are referenced by similar objects". The paper also analyzes the new similarity measure in theory and proposes an optimization algorithm which reduces the time cost from $O(kn^4)$ to $O(knl)$ where k is the number of iteration, n is the number of nodes and l is the number of edges. Experimental results demonstrate the effectiveness of the new similarity measure and efficiency of the optimization algorithm.

* 基金项目: 国家重点基础研究发展计划(973)(2014CB340402,2012CB316205); 国家自然科学基金(61272137, 61033010, 61202114); 国家社会科学基金(12&ZD220); 国家高技术研究发展计划(863)(2014AA015200); 国家高等学校学科创新引智计划

收稿时间: 2013-09-02; 修改时间: 2013-10-31; 定稿时间: 2014-01-21

Key words: random walk; similarity measure; SimRank; personalized PageRank

图数据管理与挖掘是当前数据管理的研究热点之一,基于链接的相似度度量是其中研究的一个基本问题.它在现实中有很多应用,比如链接预测[1]、协同过滤[2]、角色分析[3]、异常检测[4]和个性化推荐[5]等.现已提出许多基于链接的结点相似度度量,比如 Random Walk With restart(RWR)[6],Personalized PageRank(PPR)[7],SimRank(SR)[8]以及 Hitting time[9].其中,PPR 和 SR 最有影响力的.

PPR 算法源于促使搜索引擎巨大成功的关键技术之一的 PageRank 算法,它也应用在 Twitter 公司的朋友推荐服务中[5].PPR 近两年在学术界得到了广泛的研究,例如对它进行优化计算[10−12].RWR 是 PPR 的特殊形式:当 PPR 感兴趣的点的集合只有 1 个点时,PPR 就是 RWR.而 SR 也有很多应用——社会网络的链接预测[1]和推荐系统[13]等.最近,SR 也得到了较多的研究——图上的基于 SR 相似连接(similarity join)查询[14,15]以及 SR 的优化计算[16,17].既然 PPR 和 SR 无论在应用还是在研究当中受到这么多的关注,能不能参考这两个度量的优点,设计出一个更优秀的基于链接的相似度量呢?为此,我们首先对这些度量进行分析.

根据对两点间的路径侧重点不同,把这些度量分为两大类:

- 一类是以 SR 为代表的度量.SR 计算图中两点相似度是基于人们这样的直觉:如果两个对象同时被相同或相似的对象所引用,那么这两个对象是相似的[8].反映在网络上,实际上是考虑了如图 1 所示的路径,用随机游走模型解释为:点 a 和点 b 的相似度是指两个冲浪者分别在点 a 和点 b 出发同步逆向游走当且仅当第 1 次相遇的期望值[8].为了文中叙述方便,把如图 1 所示的路径称为逆向共同引用路径.
- 另一类是以 PPR 为代表的包括 RWR,Hitting time 的度量.它们考虑了如图 2 所示的路径,点 a 和点 b 的相似度越高表示从点 a 出发随机游走遇到点 b 的概率越大.文中把如图 2 所示的路径称为单向路径.

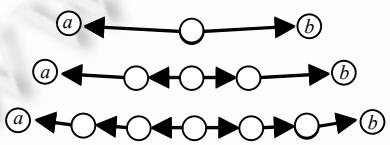

Fig.1 Reverse symmetric paths

图 1 逆向共同引用路径

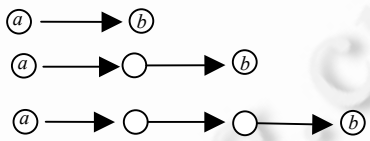

Fig.2 Unidirectional paths

图 2 单向路径

现通过一个具体例子来说明.图 3 给出了一个文献引用图,有向边$\langle a,b\rangle$表示论文 a 引用了论文 b,该文献引用图对应的部分结点对的相似度的值见表 1.

从表 1 中可以看出:(b,d)的 SR 值为 0.25,这是因为 b 和 d 之间存在一条逆向共同引用路径:$b\leftarrow c\rightarrow d$,表中的其他结点对之间不存在类似这样的逆向共同引用路径,因此,它们的 SR 值都为 0.表中第 3 列对应的是 PPR 的值,与 SR 不同的是,PPR 的值不具有对称性,即,(a,b)和(b,a)的 PPR 值是不同的;从 c 到 b 之间存在单向路径,所以对应的 PPR 值是 0.133,而其他的结点对之间不存在这样的单向路径,所以对应的 PPR 值为 0.

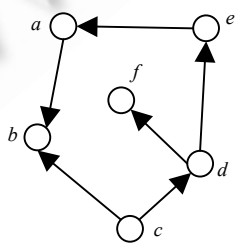

Fig.3 Citation graph

图 3 文献引用图

Table 1 Similarity scores of some node-pairs in the citation graph (Fig.3)

表 1 文献引用图(图 3)部分结点对对应的相似度

结点对	SR	PPR	SSR
(b,d)	0.25	0	0.268
(c,b)	0	0.133	0.066
(a,f)	0	0	0.047
(b,f)	0	0	0.025

从上面这个例子发现:SR 和 PPR 考虑的路径是不同的,有些结点对的 SR 值非 0 而 PPR 值却是 0;反之亦然.有些结点对的 SR 和 PPR 值同时为 0,比如(a,f)和(b,f),但是从图中可以观察到:d 引用了 e,那么 d 和 e 是相似的;而 e 和 d 又分别引用了 a 和 f,所以 a 和 f 被相似的 c 和 d 分别引用.根据“如果两个对象同时被相同或相似的对象所引用,那么这两个对象是相似的”这一原理,得出结论 a 和 f 是相似的;类似地,也可以得出 b 和 f 相似.而这样的路径也符合人们的基本认识,例如,儿子(小明)←父亲←祖父→叔叔(张三),小明和张三具有血缘关系是因为小明的父亲和张三的父亲是父子关系.但是这些本应相似的结点对对应的 SR 和 PPR 值却都为 0.

通过以上分析和讨论可知,这些度量要么侧重于逆向共同引用路径,要么侧重于单向路径.虽然实践应用表明这两大类路径对结点间的相似度都很重要,但是分析发现,它们仍然对一些符合人们基本认识且有用的路径没有考虑到,导致一些结点对相似度为 0.因此,能不能提出一个新的度量,把 PPR 和 SR 的优点有机的结合起来,同时能够考虑到更多的路径从而更加全面地体现出链接关系的本质呢?这个问题就是本文研究的出发点.

本文针对上述问题提出一个新的相似度度量.该度量有机地结合了逆向共同引用路径和单向路径,进一步加强 SR 原理,并从理论上分析该度量的合理性(第 1 节).对新公式进行变换,变换过程实际是对新公式深入分析的过程,同时发现,新度量涵盖 3 大类不同的路径.在此基础上,设计优化算法(第 2 节).然后,通过实验对该度量的质量和算法的效率进行验证(第 3 节).最后给出相关工作(第 4 节)和结论(第 5 节).

1 SuperSimRank

给定有向图 $G=\langle V,E\rangle$,$I(v)=\{u|\langle u,v\rangle\in E\}$表示图中结点 v 的入度邻居集合,$I_i(v)(1\leqslant i\leqslant|I(v)|)$为该集合中的一个元素;$O(v)=\{u|\langle v,u\rangle\in E\}$表示图中结点 v 的出度邻居集合,$O_i(v)(1\leqslant i\leqslant|O(v)|)$为该集合中的一个元素.为了方便理解,首先对 SR 和 PPR 做一个简介.

1.1 SR和PPR简介

SR 相似度公式 $S(a,b)=\begin{cases}\dfrac{c}{|I(a)||I(b)|}\displaystyle\sum_{i}^{|I(a)|}\sum_{j}^{|I(b)|}S(I_i(a),I_j(b)), & a\neq b\\ 1, & a=b\end{cases}$ 规定:结点对结点本身的相似度最大,例如 $S(a,a)=1$.从公式中可以知道:两个结点的相似度,是它们的入度邻居之间相似度的平均值.关于 SR 的详细信息可参见文献[8].

与 PageRank 不同的是,PPR 强调的是个性化,随机冲浪者以概率 c 沿着链接游走,在每一步游走的同时,以概率 $1-c$ 跳到他感兴趣的点,如果他感兴趣的点的集合只包含 1 个点 q,那么对应的 PPR 公式为

$$r(q,v)=(1-c)\sum_{\tau:q\to v}p(\tau)c^{l(\tau)}.$$

这里,τ表示点 q 到 v 的单向路径:$(q,w_1,\ldots,w_n,v)$, $p(\tau)=\dfrac{1}{|O(q)|}\displaystyle\prod_{i=1}^{n}\frac{1}{|O(w_i)|}$ 表示该单向路径的概率,$l(\tau)$表示该路径的长度;$r(q,v)$的值表示在点 q 观点下,它与 v 的相似度值.

而在实际中,利用公式 $r_k(q,v)=(1-c)\displaystyle\sum_{\substack{\tau:q\to v\\ l(\tau)\leqslant k}}p(\tau)c^{l(\tau)}$ 来快速计算,详情见文献[7].

1.2 SuperSimRank公式

文中把点 a 和点 b 的 SuperSimRank 值表示为 $M(a,b)$,并把 SuperSimRank 简记为 SSR,对应的公式为

$$M_{k+1}(a,b)=\frac{c}{|I(a)||I(b)|}\sum_{i=1}^{|I(a)|}\sum_{j=1}^{|I(b)|}M_k(I_i(a),I_j(b))+T_{k+1}(a,b) \tag{1}$$

这里, $T_{k+1}(a,b)=(1-c)\left(\sum_{\tau:a\to b,\ l(\tau)\leqslant k+1}c^{l(\tau)}p(\tau)+\sum_{\tau:b\to a,\ l(\tau)\leqslant k+1}c^{l(\tau)}p(\tau)\right)\Big/2$.

在上述公式中,c 为常数取值 0.5,且 $a\neq b,\tau.a\to b$ 为 a 到 b 的单向路径,$l(\tau)$为该路径的长度,$p(\tau)$是为 a 到 b 的单向路径的概率.规定:结点对结点本身的相似度最大为 1.

规定初始值 $M_0(a,b)$:当 $a=b$ 时为 1,否则为 0.

SSR 对应的公式(1)是迭代公式,公式分为两部分:第 1 部分与 SR 公式形式相同;第 2 部分 T_{K+1} 实际是 a 到 b 与 b 到 a 的 PPR 值的平均值,取平均值的原因是为了度量值具有对称性,即 a 与 b 和 b 与 a 的 SSR 值相同.所以,T_{K+1} 的值同时参考了 a 和 b 的观点.

公式(1)给人的错觉就是简单地把 SR 和 PPR 求和,其实不然.公式(1)除了考虑了图 1 的逆向共同引用路径和图 2 的单向路径以外,还考虑了如图 4 所示的路径.在图 4 中,路径长度分别为 3 和 4,椭圆虚线框起来的路径对应的值包含在上一次迭代的 T_K 中,而 T_K 值加在对应的 M_K 上,因此在下一次迭代中,通过公式(1)的第 1 部分把相似性扩散到 a 和 b 上.这加强了 SR 的原理:两个对象是相似的,是因为它们被相似的对象所引用.与 SR 不同的是,这里相似的对象是通过单向路径得到的(图中虚线内的部分).

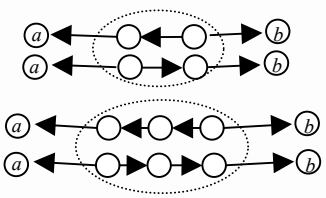

Fig.4 New paths that SuperSimRank considers

图 4 SuperSimRank 所额外考虑的路径

综上所述,SSR 不仅考虑了逆向共同引用路径和单向路径,而且扩展了 SR 原理,考虑了更多的 SR 和 PPR 之前没有考虑过的路径.表 1 的第 4 列就是对应的 SSR 值,可以看出,公式(1)很好地解决了之前存在的问题.

1.3 SuperSimRank性质

公式(1)是一个迭代的公式,它作为一个度量必须保证是收敛有极限,因此用如下定理描述:

定理 1. SSR 迭代公式(公式(1))具有以下性质:

1. 对称性:$M_k(a,b)=M_k(b,a)$.
2. 单调性:$M_k(a,b)\leqslant M_{k+1}(a,b)$.
3. 极限存在且唯一.

证明:

1. 从公式定义可以看出,显然其具有对称性.
2. 单调性:$M_{k+1}(a,b)$在 $M_k(a,b)$考虑的路径基础上,又考虑了长度为 k+1 的路径,因此,$M_{k+1}(a,b)$比 $M_k(a,b)$多了长度为 k+1 的路径的贡献值.第 2.2 节中的公式(4)很好地反映了 SSR 是单调递增的.
3. 极限存在且唯一:假设 $a\neq b$:

- k=1:

$$M_1(a,b)=\frac{c}{|I(a)||I(b)|}\sum_{i=1}^{|I(a)||I(b)|}\sum_{j=1}^{|I(b)|}M_0(I_i(a),I_j(b))+T_1(a,b)\leqslant c+T_1=c+c(1-c)<2c.$$

因为文中 $M_0(c,d)$(当 $c=d$ 时)取值为 1,表示自己对自己完全相似,因此,为了使 $M_1(a,b)$小于 1,对 c 的取值是 0.5.在以下的证明中,假设 $M_k(I_i(a),I_j(b))$能够取到最大值 1,而且单向路径的概率是小于 1,所以有以下不等式成立:

- k=2:

$$M_2(a,b)=\frac{c}{|I(a)||I(b)|}\sum_{i=1}^{|I(a)||I(b)|}\sum_{j=1}^{|I(b)|}M_1(I_i(a),I_j(b))+T_2(a,b)\leqslant c+T_2=c+(1-c)(c+c^2)=c+c-c^3<2c;$$

- k=n:

$$M_n(a,b)=\frac{c}{|I(a)||I(b)|}\sum_{i=1}^{|I(a)||I(b)|}\sum_{j=1}^{|I(b)|}M_{n-1}(I_i(a),I_j(b))+T_n(a,b)\leqslant c+T_n=c+(1-c)\frac{c(1-c^n)}{1-c}=c+c-c^{n+1}<2c.$$

因此,数列$\{M_k(a,b)\}$单调递增有上界,所以,数列收敛且极限唯一.证毕. □

公式(1)的上述属性确保了它能够作为相似度的度量.证明中发现 $M_k(a,b)<2c$,因此,为了使度量值不超过 1,文中对 c 取常数 0.5.

2 SuperSimRank 计算算法

2.1 Naïve方法

直接计算 SSR 度量实际上是在每次迭代中对公式(1)中的两部分分别计算,然后求和.假设图 G 有 n 个结点,每个结点的平均入度(出度)为 D,共有 n^2 个结点对.第 1 部分和 SR 形式一致,因此,根据文献[18]计算第 1 部分值的时间复杂度是 $O(n^2D^2)$,最坏情况是 $O(kn^4)$;第 2 部分实际上是求单向路径的概率,我们可以在上一次迭代的长度为 k 的路径(最多有 n^2)的基础上再随机游走一步,可得到当前路径的值,那么第 2 部分的时间代价是 $O(n^2D)$.因此,总的时间复杂度是 $O(kn^2D^2)$,k 为迭代次数.当 D 的值接近 n 时,时间复杂度变为 $O(kn^4)$.Naïve 方法代价太高,需要更好的方法去计算它.

2.2 优化方法

为了引入优化算法,我们将公式(1)变换为如下形式:

$$M'_{k+1}(a,b)=\sum_{\substack{l(\tau')=1\\ \tau':(x,y)\to(a,b)}}^{k+1-d}c^{l(\tau')}p(\tau')+\frac{1-c}{2}\sum_{\substack{l(\tau)=1\\ \tau:a\to b\cup\tau:b\to a}}^{k+1}c^{l(\tau)}p(\tau)\tag{2}$$

这里,当 $x=y$ 时,τ'表示冲浪者分别从 a 和 b 逆向同步游走且第 1 次在点 x 相遇,路径长度 $l(\tau')$为逆向同步游走步数且 d 为 0;否则,(x,y)表示 x 到 y 或 y 到 x 的单向路径且对应路径长度是 d;同时,$(x,y)\to(a,b)$表示冲浪者分别从 a 和 b 逆向同步游走且同时分别抵达点 x 和 y,逆向游走步数是 m,则对应的 $l(\tau')$取值是 m,d 和 m 满足 $d+m\leqslant k+1$.τ'如图 5 所示,图中只画了从 x 到 y 的单向路径,对应的路径概率为

$$p(\tau')=\frac{t_d(x,y)}{|I(a)||I(b)|}\left(\prod_{t=1}^{m-1}\frac{1}{|I(a_t)||I(b_t)|}\right)\tag{3}$$

这里,$t_d(x,y)=\dfrac{1-c}{2}\displaystyle\sum_{\substack{l(\tau)=d\\ \tau:x\to y\cup\tau:y\to x}}c^{l(\tau)}p(\tau)$.

下面通过引理 1 来说明公式(1)和公式(2)相同.

引理 1. SuperSimRank 迭代公式(1)与公式(2)相同: $M_{k+1}(a,b)=M'_{k+1}(a,b)$.

详细证明见附录 1.

由上述公式变换可知,新度量涵盖了 3 大类路径:逆向共同引用路径、单向路径和如图 5 所示的路径.图 5 所示路径加强了 SR 的原理:两个对象是相似的,是因为它们被相似的对象所引用;同时,这也符合人们的基本

认识.

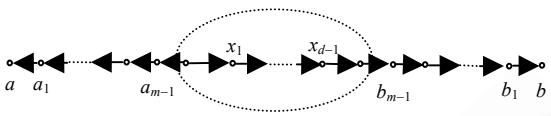

Fig.5 Paths of τ' ($x\neq y$)

图 5 τ'对应的路径($x\neq y$)

利用公式(2)可以把 SSR 迭代公式写成

$$M_{k+1}(a,b)=M_k(a,b)+\sum_{\substack{(d+l(\tau'))=k+1\\ \tau':(x,y)\to(a,b)}}c^{l(\tau')}p(\tau')+\frac{1-c}{2}\sum_{\substack{l(\tau)=k+1\\ \tau:a\to b\cup\tau:b\to a}}c^{l(\tau)}p(\tau) \tag{4}$$

其中,d 表示从 x 到 y 或从 y 到 x 的单向路径的长度,如果 $x=y$,则 d 为 0.

从公式(4)知:第 k+1 步的 SSR 值实际上就是把对应长度为 k+1 的路径的贡献值累加在上次的结果上(这里,我们把满足(d+l(τ'))=k+1 的路径也称为长度为 k+1 的路径),优化方法就是基于公式(4)的,从公式中可知,计算第 k+1 步 SSR 值的关键就是如何有效地计算出对应长度为 k+1 的路径的贡献值.

计算长度为 k+1 的路径的值的有效方式是,在长度为 k 的路径基础上再走一步即可.为此,我们采取以下措施:

1. 每次保留上一步长度为 k 的路径τ'和单向路径;
2. 原来采取逆向游走的,现采取同向游走,原因是方便在长度为 k 的路径的基础上得到长度为 k+1 的路径的值;
3. 路径τ':(x,y)→(a,b)只用(a,b)表示(规定 $a\leqslant b$),目的是合并路径:每一次迭代把结点对(a,b)相同的τ'合并在一起,同时,把当前的从 a 到 b 或从 b 到 a 单向路径加在(a,b)上.

假设 e,f 分别是 a,b 入度邻居结点,从 e,f 顺向出发走一步可分别到达 a,b,长度为 k 的合并后单向路径和路径τ'分别表示为

$$u_k(e,f)=\sum_{\substack{l(\tau)=k\\ \tau:e\to f}}c^{l(\tau)}p(\tau);$$

$$w_k(e,f)=\sum_{\substack{l(\tau')=k\\ \tau':(x,y)\to(e,f)}}c^{l(\tau')}p(\tau')+\frac{1-c}{2}(u_k(e,f)+u_k(f,e)).$$

那么,

$$u_{k+1}(a,b)=c\sum_{f\in I(b)}\frac{u_k(a,f)}{|O(f)|};$$

$$w_{k+1}(a,b)=\frac{c}{|I(a)||I(b)|}\sum_{\substack{e\in I(a)\\ f\in I(b)}}w_k(e,f)+\frac{1-c}{2}(u_{k+1}(a,b)+u_{k+1}(b,a)).$$

这时,公式(4)可以写成

$$M_{k+1}(a,b)=M_k(a,b)+w_{k+1}(a,b) \tag{5}$$

通过以上分析得到了优化算法,见算法 1.

算法 1 是优化算法,第 1 行~第 10 行是计算 u_1 和 w_1 的值;第 13 行~第 17 行假设 e,f 分别是 a,b 入度邻居结点,每一行中是对所有可能的结点对进行相应的操作,其中,第 15 行和第 16 行的计算代价最高,$w_m(e,f)$($pw_m(f,a)$)共有 n^2 个,对 e(f)的每个出度邻居 a(b),计算 $pw_m(f,a)$($w_{m+1}(a,b)$),所以时间代价是 $O(n^2D)$.因此,算法 1 的时间代价是 $O(knl)$.这里,$l=nD$.

输入:图 G,最大的迭代次数 K,常数 c.
输出:SuperSimRank 值.
1: FOR 每一个点 $v\in V(G)$ DO:
2: For i=0 To $|O(v)|$ DO:
3: $u_1(v,O_i(v))=\frac{c}{|O(v)|}$
4: For j=i+1 To $|O(v)|$ DO:
5: IF $O_i(v)<O_j(v)$ THEN:
6: $w_1(O_i(v),O_j(v))+=\frac{c}{|I(O_i(v))||I(O_j(v))|}$;
7: ELSE:
8: $w_1(O_i(v),O_j(v))+=\frac{c}{|I(O_i(v))||I(O_j(v))|}$;
9: END FOR
10: END FOR
11: m=1
12: WHILE $m\leqslant K$ DO:
13: 合并: $w_m(e,f)+=\frac{1-c}{2}(u_m(e,f)+u_m(f,e))$
14: 计算:$M_m(a,b)=M_{m-1}(a,b)+w_m(a,b)$
15: 计算: $pw_m(f,a)+=\frac{w_m(e,f)}{|I(a)|}$
16: 计算: $w_{m+1}(a,b)+=\frac{c\times pw_m(f,a)}{|I(b)|}$
17: 计算: $u_{m+1}(e,b)+=c\times\frac{u_m(e,f)}{|O(f)|}$
18: m++
19: END WHILE

Fig.6 Algorithm 1

图 6 算法 1

3 实 验

实验的运行环境为 i7-2620M CPU,8GB 内存和 Windows 7 操作系统.文中算法采用 C++实现.

实验主要考察 3 个方面:一是评估 SSR 度量的质量,即通过与 SR 和 PPR 相比较,SSR 度量是不是具有优势;二是效率的比较,将优化算法和 Naïve 算法进行比较;三是考察 SSR 度量的收敛速度.

为了使所有实验都能重复实现且真实可信,所有的真实数据都可以从网上下载,生成数据是用开源软件生成的.

真实数据集 1:真实数据集采用 CiteSeer 和 Cora 数据集(http://www.cs.umd.edu/projects/linqs/projects/lbc/index.html),数据集记录了文献引用关系,组成了文献引用网络,数据集中的论文已根据主题分了类,具体信息见表 2;CiteSeer 共有 3 312 个结点,4 715 条有向边;而 Cora 共有 2 708 个结点,5 429 条有向边.在文献引用关系中,如果论文 i 引用了论文 j,那么存在一条有向边$\langle i,j\rangle$.

文献引用图具有一定的代表性:(1) 在文献引用图中,论文之间引用是因为它们有某些共同点或有借鉴之处,这与其他网络类似.其他网络结点间具有关系也是因为它们具有某些相似处或借鉴之处,例如社交网络用户相互关注是因为他们有某些共同兴趣或某些方面吸引到对方.(2) 在实验中,两个文献引用图在对主题的划分粒度上不太一样,CiteSeer 划分为大方向的主题,而 Cora 划分为小而具体的主题.这两个真实数据集具有不同的特点.

真实数据集 2:Twitter 社会网络数据集(http://snap.stanford.edu/data/egonets-Twitter.html).在这个数据集中,分别选择了编号为 2363991 和 819800 的 Twitter 图.图中如果用户 a 关注用户 b(a follow b),就存在一条从 a 到 b 的有向边.编号为 2363991 的 Twitter 图共有 214 个点,4 462 条边,这些点被划分在 9 个不同的圈子(circles),有

的点可以属于多个圈子(下同).编号为 819800 的 Twitter 图共有 91 个点,2 400 条边,这些点被划分在 10 个不同的圈子.

Table 2 Distribution of papers over topics

表 2 基于主题的论文分布

CiteSeer 主题	对应的论文数	Cora 主题	对应的论文数
Agents	596	Neural_Networks	818
IR	668	Rule_Learning	180
DB	701	Reinforcement_Learning	217
AI	249	Probabilistic_Methods	426
HCI	508	Theory	351
ML	590	Genetic_Algorithms	418
–	–	Case_Based	298
总计	3 312	总计	2 708

这 4 个真实数据集是用于评估 SSR 质量的.我们采用文献[19]中类似的公式来进行评估:

$$precision_{A,N}(v)=\frac{|top_{A,N}(v)\cap similar(v)|}{|top_{A,N}(v)|},$$

$$recall_{A,N}(v)=\frac{|top_{A,N}(v)\cap similar(v)|}{N},$$

$$Fscore_{A,N}(v)=2\frac{precision_{A,N}(v)\times recall_{A,N}(v)}{precision_{A,N}(v)+recall_{A,N}(v)}.$$

这里,符号 $top_{A,N}(v)$表示用算法 A 求出关于结点 v 的 N 个最相似的点的集合,$similar(v)$作为 ground truth,表示与 v 属于同一个主题的论文的集合或与 v 属于同一个圈子的用户的集合.

图 7~图 12 是分别在文献引用图这两个数据集上各随机取了 200 个点后算出的平均值.从图中可以看出:SSR 明显好于 SR 和 PPR;PPR 在 Cora 数据集上优于 SR,SR 在 CiteSeer 数据集上优于 PPR.

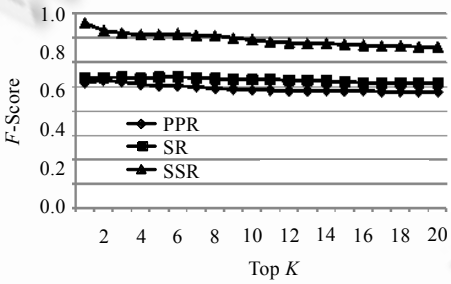

Fig.7 Average precision over Citeseer

图 7 在 CiteSeer 上的平均 precision

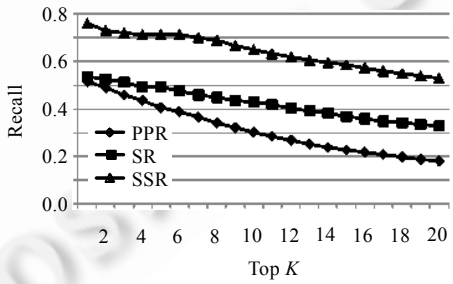

Fig.8 Average recall over Citeseer

图 8 在 CiteSeer 上的平均 recall

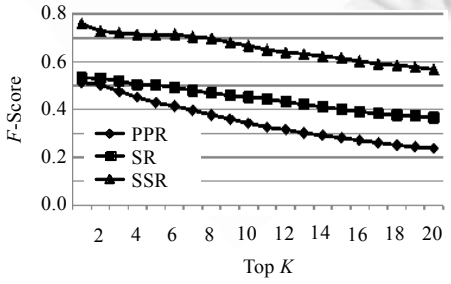

Fig.9 Average *F*-score over Citeseer

图 9 在 CiteSeer 上的平均 *F*-score

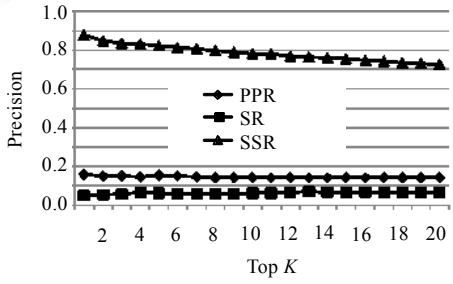

Fig.10 Average precision over Cora

图 10 在 Cora 上的平均 precision

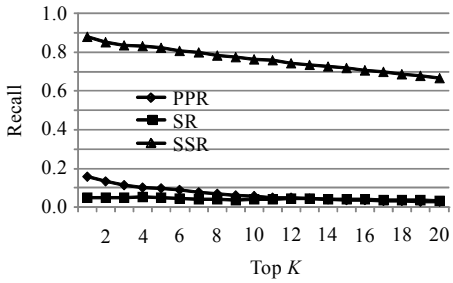

Fig.11 Average recall over Cora

图 11 在 Cora 上的平均 recall

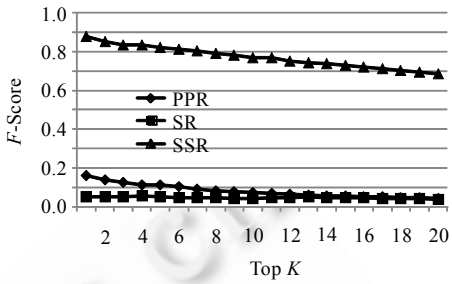

Fig.12 Average *F*-score over Cora

图 12 在 Cora 上的平均 *F*-score

因为 Twitter 图上有的点可以属于多个圈子,所以为了计算方便,我们找出所有只属于 1 个圈子的点.编号为 2363991 和 819800 的 Twitter 图中,这样的点的数目分别为 158 和 32,图 13 和图 14 就是这些符合条件的所有点算出的平均值.因为两个数据集上 recall,*F*-Score 变化趋势与对应的 precision 图一致,所以为了节省空间没有列出.从图中可以看出,在两个不同的数据集上,SSR 均优于 PPR 和 SR.

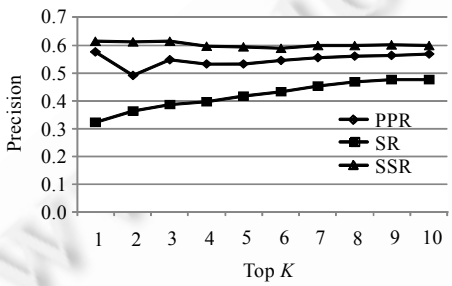

Fig.13 Average precision over Twitter graph of No.2363991

图 13 在编号为 2363991 的 Twitter 图上的平均 precision

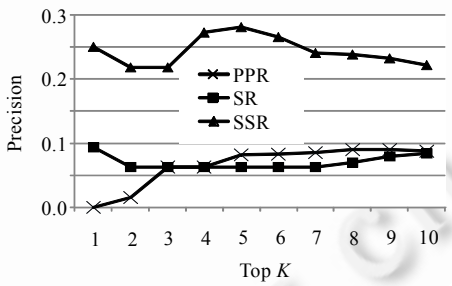

Fig.14 Average precision over Twitter graph of No.819800

图 14 在编号为 8198001 的 Twitter 图上的平均 precision

以上实验迭代次数都是 20,经验证,在这些数据集上,SR 和 PPR 经过 20 次迭代都已收敛.通过在 4 个不同真实数据集上的比较,尽管 SR 和 PPR 在不同数据集上变化较大,然而 SSR 整体上始终优于 SR 和 PPR.

本文的着眼点是度量的准确性,而不是计算速度.优化后,三者的计算复杂度一致,但 SSR 的值包含了对应的 PPR 和 SR 的值,理论上,在相同迭代次数下,计算 SSR 比计算 PPR 和 SR 要稍慢一些,采用文中介绍的优化方法对 3 个度量在 Cora 和 CiteSeer 上进行计算,图 15 和图 16 是在不同迭代次数下,三者的对应运行时间,在最坏情况下,SSR 要比另外两个慢 0.5s,而这样的差距是很小的.因此,SSR 的计算速度是可以接受的.

然后进行效率的比较.优化算法和 Naïve 算法除了在上述文献引用图的两个真实数据集外(因为 Twitter 图结点数比较少,所以以下实验没有采用这些数据集),又人工生成一系列数据进行了比较.所有的迭代次数都是 8.

图 17 是在这两个真实数据集上运行时间的比较,图 18 是在不同生成数据集(使用 networkX(http://networkx.lanl.gov/index.html#)生成,下同)上的运行时间,在这些数据集中,结点的平均出度是 3.图 19 是 1 000 个结点上不同平均出度下的运行时间.从图中可以看出:无论何种情况,优化算法都明显快于 Naïve 方法.

接下来,在生成数据和真实数据集上考察 SSR 的收敛性.这里,通过当前迭代的值与上次迭代值的差值来表示收敛性.如果差值为 0,则说明迭代并没有使值发生变化.在实验中,我们选取了所有在第 1 次迭代后 SSR 值非 0 的结点对,然后在每一次迭代中,利用这些结点对算出一个平均 SSR 值,求出相应的差值.图 20 是我们在不同生成数据集上的一组实验.这些数据集结点数都是 1 000,出度分别从 3 增加到 7.从图中可以看出,第 8 次迭代的

结果值与第 9 次值的差值已经很小了.图 21 是在两个真实数据集上的结果,从图中我们可以看出,只需迭代 5 次就足够了.因此,我们建议一般情况下迭代 8 次即可.

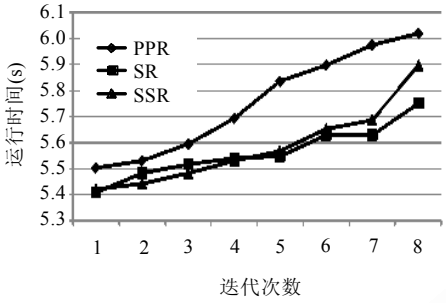

Fig.15 Runtime of SSR, SR and PPR over Cora

图 15 Cora 上的 SSR,SR 和 PPR 的运行时间

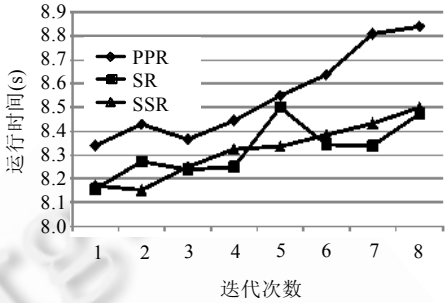

Fig.16 Runtime of SSR, SR and PPR over CiterSeer

图 16 CiterSeer 上的 SSR,SR 和 PPR 的运行时间

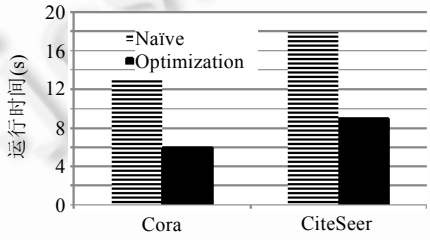

Fig.17 Runtime of optimize algorithm versus runtime of Naïve algorithm over Cora and CiteSeer

图 17 优化算法和 Naïve 方法在 Cora 和 CiteSeer 上的运行时间比较

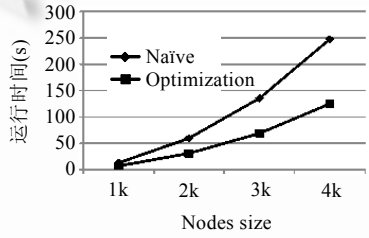

Fig.18 Runtime on generated dataset of varying size

图 18 不同规模的生成数据集上的运行时间

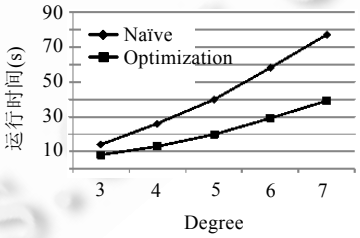

Fig.19 Runtime on dataset of varying average degree

图 19 平均度数不同的数据集上的运行时间

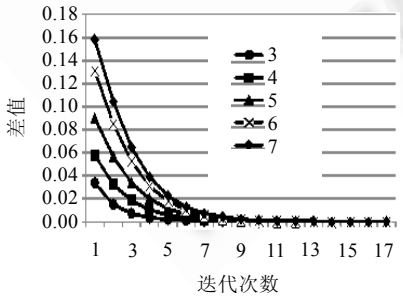

Fig.20 SSR convergence over generated dataset of varying degree

图 20 SSR 在度数不同的生成数据集上的收敛情况

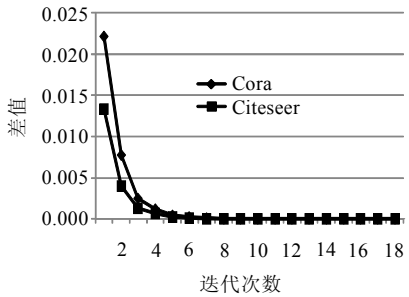

Fig.21 SSR convergence over real dataset

图 21 SSR 在真实数据集上的收敛情况

4 相关工作

图数据管理与挖掘得到了广泛的研究,例如最短距离(shortest path)[20]、可达性(reachability)[21]、图聚类(graph clustering)[22]和图模式匹配(graph pattern matching)[23]等.基于链接的相似度度量,是图数据挖掘中的基本问题之一[19].

相似度度量可以划分为两大类[18]:基于文本(内容)的和基于链接的度量.本文提出的 SSR 以及 RWR,PPR,SR 和 Hitting time 都是基于链接的相似度度量.

除了上述度量之外,P-rank[24]是 SR 的扩展,但它同时考虑了入度和出度链接.然而,如果两点之间的路径只是如图 2 所示那样,那么 P-rank 会把这样结点对的相似性也视为 0;而我们的 SSR 却能算出这样的结点对的相似性.另外, SimRank++[2]增加了一个叫 evidence 的权重.MatchSim[19]定义两个结点的相似度,是通过它们的最大匹配的相似邻居对的平均相似度来定义的.这种相似度定义的最大问题是运算代价比较高.SimFusion[25]和 PathSim[26]都是定义在异构网络上的相似度,而我们的 SSR 和 SR 都是定义在同构网络上的.最近提出的 RoundTripRank[27]与上述度量不同的是,把 specificity 和 importance 有机结合在一起.Lee 等人提出了一种算法,求与给定结点 SimRank 值最高的 top-k 查询[28].

有许多工作集中在如何有效而快速地计算 PPR 和 SR.Jeh 等人提出了计算 PPR 的可扩展方案:任何一个 personalized PageRank vectors 都可以表达成基本向量的线性组合[7].而比较快的方法是通过蒙特卡洛的方法来模拟随机游走,从而估算 PPR 的值[29,30].文献[10]通过基于上下界的方法来避免没必要的计算开支,去求给定结点的 top-k 相关的点.文献[12]进一步提出了基于 PPR 的支持即时查询的更为快速的 Top-k 查询.

文献[31]提出了一个基于蒙特卡洛计算 SR 的框架:预计算了随机的 fingerprints 的索引,查询时根据这些 fingerprints 来估算 SR 值.该方法是基于外存的算法.文献[18]提出了一种有效的计算 SR 的算法,使得算法性能从最坏情况 $O(kn^4)$提高到 $O(knl)$;同时给出了一种阈值过滤的方法,能够过滤许多小而无用的值,从而提高计算速度.Li 等人提出了增量更新相似度值估量算法[32].文献[33]提出了无需计算其他结点对的 SR 值,能够直接计算出指定结点对的 SR 值.

5 结 论

本文首先分析了两种代表性的图上相似度度量 SR 和 PPR.这两种度量建立在图上的不同路径上.在此基础上提出了一个新的度量 SSR,它不仅同时考虑了 SR 和 PPR 的优点,而且加强了 SR 的原理——两个对象相似是因为它们引用了相似的对象.然后,对 SSR 进行理论分析.接着,对新公式进行了变换,变换过程实际上是对新公式深入分析的过程.在此基础上设计了优化算法,使得算法性能从最坏情况 $O(kn^4)$提高到 $O(knl)$.最后,通过实验验证了 SSR 优于 SR 和 PPR,同时验证了优化算法的有效性.

大图上的 SSR 优化计算不是本文的重点.未来我们的工作主要是进行这方面的研究.

致谢 感谢审稿专家和编辑老师对本文提出的宝贵意见和建议.

References:

[1] Liben-Nowell D, Kleinberg J. The link prediction problem for social networks. Journal of the American Society for Information Science and Technology, 2007,58(7):1019−1031. [doi: 10.1002/asi.20591]

[2] Antonellis I, Molina HG, Chang CC. Simrank++: Query rewriting through link analysis of the click graph. Proc. of the VLDB Endowment, 2008,1(1):408−421. [doi: 10.14778/1453856.1453903]

[3] Jin R, Lee VE, Hong H. Axiomatic ranking of network role similarity. In: Proc. of the 17th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining (KDD 2011). New York: ACM Press, 2011. 922−930. [doi: 10.1145/2020408.2020561]

[4] Gyöngyi Z, Garcia-Molina H, Pedersen J. Combating Web spam with trustrank. In: Proc. of the 30st Int'l Conf. on Very Large Data Bases (VLDB 2004). New York: ACM Press, 2004. 576−587.

[5] Gupta P, Goel A, Lin J, Sharma A, Wang D, Zadeh R. WTF: The who to follow service at Twitter. In: Proc. of the 22Nd Int'l Conf. on World Wide Web (WWW 2013). New York: ACM Press, 2013. 505−514.

[6] Fujiwara Y, Nakatsuji M, Onizuka M, Kitsuregawa M. Fast and exact top-*k* search for random walk with restart. Proc. of the VLDB Endowment, 2012,5(5):442−453. [doi: 10.14778/2140436.2140441]

[7] Jeh G, Widom J. Scaling personalized Web search. In: Proc. of the 12th Int'l Conf. on World Wide Web (WWW 2003). New York: ACM Press, 2003. 271−279. [doi: 10.1145/775152.775191]

[8] Jeh G, Widom J. SimRank: A measure of structural-context similarity. In: Proc. of the 8th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining (KDD 2002). New York: ACM Press, 2002. 536−543. [doi: 10.1145/775047.775126]

[9] Sarkar P, Moore AW, Prakash A. Fast incremental proximity search in large graphs. In: Proc. of the 25th Internet Conf. on Machine Learning (ICML 2008). New York: ACM Press, 2008. 896−903. [doi: 10.1145/1390156.1390269]

[10] Fujiwara Y, Nakatsuji M, Yamamuro M, Shiokawa H, Onizuka M. Efficient personalized PageRank with accuracy assurance. In: Proc. of the 18th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining (KDD 2012). New York: ACM Press, 2012. 15−23. [doi: 10.1145/2339530.2339538]

[11] Zhu F, Fang Y, Jing Y. Incremental and accuracy-aware personalized PageRank through scheduled approximation. Proc. of the VLDB Endowment, 2013,6(6):481−492. [doi: 10.14778/2536336.2536348]

[12] Fujiwara Y, Nakatsuji M, Yamamuro M, Shiokawa H, Onizuka M. Efficient ad-hoc search for personalized PageRank. In: Proc. of the 2013 ACM SIGMOD Int'l Conf. on Management of Data. New York: ACM Press, 2013. 445−456. [doi: 10.1145/2463676.2463717]

[13] Abbassi Z, Mirrokni VS. A recommender system based on local random walks and spectral methods. In: Proc. of the 9th WebKDD and 1st SNA-KDD 2007 Workshop on Web Mining and Social Network Analysis. New York: ACM Press, 2007. 102−108. [doi: 10.1007/978-3-642-00528-2_8]

[14] Sun L, Cheng R, Li X, Cheung DW, Han J. On link-based similarity join. Proc. of the VLDB Endowment, 2011,4(11):714−725.

[15] Zheng W, Zou L, Feng Y, Chen L, Zhao D. Efficient SimRank-based similarity join over large graphs. Proc. of the VLDB Endowment, 2011,4(11):493−504. [doi: 10.14778/2536349.2536350]

[16] Fujiwara Y, Nakatsuji M, Shiokawa H, Onizuka M. Efficient search algorithm for SimRank. In: Proc. of the 29th Int'l Conf. on Data Engineering (ICDE 2013). Washington: IEEE Computer Society, 2013. 589−600. [doi: 10.1109/ICDE.2013.6544858]

[17] Yu W, Lin X, Zhang W. Towards efficient SimRank computation on large networks. In: Proc. of the 29th Int'l Conf. on Data Engineering (ICDE 2013). Washington: IEEE Computer Society, 2013. 601−612. [doi: 10.1109/ICDE.2013.6544859]

[18] Lizorkin D, Velikhov P. Accuracy estimate and optimization techniques for simrank computation. The VLDB Journal, 2010,19(1):45−66. [doi: 10.1007/s00778-009-0168-8]

[19] Lin ZJ, Lyu MR, King I. MatchSim: A novel similarity measure based on maximum neighborhood matching. Knowledge and Information Systems, 2012,32(1):141−166. [doi: 10.1007/s10115-011-0427-z]

[20] Xiao Y, Wu W, Pei J, Wang W, He Z. Efficiently indexing shortest paths by exploiting symmetry in graphs. In: Proc. of the 12th Int'l Conf. on Extending Database Technology (EDBT 2009). New York: ACM Press, 2009. 493−504. [doi: 10.1145/1516360.1516418]

[21] Triβl S, Leser U. Fast and practical indexing and querying of very large graphs. In: Proc. of the 2007 ACM SIGMOD Int'l Conf. on Management of Data. New York: ACM Press, 2007. 845−856. [doi: 10.1145/1247480.1247573]

[22] Schloegel K, Karypis G, Kumar V. Parallel static and dynamic multi-constraint graph partitioning. Concurrency and Computation: Practice and Experience, 2002,14(3):219−240. [doi: 10.1002/cpe.605]

[23] Fan W, Li, Ma S, Tang N, Wu Y. Graph pattern matching: From intractable to polynomial time. Proc. of the VLDB Endowment, 2010,3(1):264−275.

[24] Zhao P, Han J, Sun Y. P-Rank: A comprehensive structural similarity measure over information networks. In: Proc. of the 18th ACM Conf. on Information and Knowledge Management. New York: ACM Press, 2009. 553−562. [doi: 10.1145/1645953.1646025]

[25] Xi WS, Fox EA, Fan B. SimFusion: Measuring similarity using unified relationship matrix. In: Proc. of the 28th Annual Int'l ACM SIGIR Conf. on Research and Development in Information Retrieval. New York: ACM Press, 2005. 130−137. [doi: 10.1145/1076034.1076059]

[26] Sun Y, Han J. Pathsim: Meta path-based top-*k* similarity search in heterogeneous information networks. Proc. of the VLDB Endowment, 2011,4(11):992−1003.

[27] Fang Y, Chang KCC, Lauw HW. RoundTripRank: Graph-Based proximity with importance and specificity. In: Proc. of the 29th Int'l Conf. on Data Engineering (ICDE 2013). Washington: IEEE Computer Society, 2013. 613−624. [doi: 10.1109/ICDE.2013.6544860]

[28] Lee P, Lakshmanan LVS, Yu JX. On top-*k* structural similarity search. In: Proc. of the 28th Int'l Conf. on Data Engineering (ICDE 2012). Washington: IEEE Computer Society, 2012. 774−785. [doi: 10.1109/ICDE.2012.109]

[29] Fogaras D, Csalogany K, Racz B, Sarlos T. Towards scaling fully personalized PageRank: Algorithms, lower bounds, and experiments. Internet Mathematics, 2005,17(2):333−358. [doi: 10.1080/15427951.2005.10129104]

[30] Bahman B, Chakrabarti K, Xin D. Fast personalized PageRank on MapReduce. In: Proc. of the 2011 ACM SIGMOD Int'l Conf. on Management of Data. New York: ACM Press, 2011. 973−984. [doi: 10.1145/1989323.1989425]

[31] Fogara D, Racz B. Practical algorithms and lower bounds for similarity search in massive graphs. IEEE Trans. on Knowledge and Data Engineering, 2007,19(5):585−598. [doi: 10.1109/TKDE.2007.1008]

[32] Li C, Han J, He G. Fast computation of simrank for static and dynamic information networks. In: Proc. of the 13th Int'l Conf. on Extending Database Technology (EDBT 2010). New York: ACM Press, 2010. 465−476. [doi: 10.1145/1739041.1739098]

[33] Li P, Liu H, Yu JX, He J, Du X. Fast single-pair simrank computation. In: Proc. of the SIAM Int'l Conf. on Data Mining. Philadelphia: SIAM, 2010. 571−582.

附录 1. 引理 1 的证明

证明:显然, $M_1(a,b)=M_1'(a,b)$. 假设 $M_k(a,b)=M_k'(a,b)$ 成立,证明当 k+1 时公式也相同.

两个公式对应的第 2 部分相同,现证明对应的第 1 部分也相同,即

$$\frac{c}{|I(a)||I(b)|}\sum_{i=1}^{|I(a)|}\sum_{j=1}^{|I(b)|}M_k(I_i(a),I_j(b))=\sum_{\substack{l(\tau')=1\\ \tau':(x,y)\to(a,b)}}^{k+1-d}c^{l(\tau')}p(\tau').$$

我们把 τ'路径分为两大类:

- 一类是 $x\neq y$,且 $l(\tau')$取值是 1.也就是说,x,y 分别属于 $I(a)$和 $I(b)$,分别从 a 和 b 逆向走一步就到达了 x 和 y,(x,y)表示从 x 到 y 或从 y 到 x 的单向路径且路径长度是 $d(1\leqslant d\leqslant k)$;
- 剩下的路径划为另一类,我们记为$(x,y)\to(I(a),I(b))\to(a,b)$.

因此,无论哪一类路径,τ'都可以看作由两部分组成:

- 一部分是从 a 和 b 逆向走一步就到达了它们对应的入度邻居结点,这部分对 $p(\tau')$的贡献值是
$$\frac{1}{|I(a)||I(b)|}.$$
- 剩下的部分要么是单向路径(x,y),由公式(3)可知,它对 $p(\tau')$的贡献值是 $t_d(x,y)$;要么是$(x,y)\to(I(a),I(b))$,对 $p(\tau')$的贡献值是 $\sum_{\substack{l(\tau')=1\\ \tau':(x,y)\to(I(a),I(b))}}^{k}c^{l(\tau')}p(\tau')$.

所以,把路径分为两部分后,得到如下式子:

$$\sum_{\substack{l(\tau')=1\\ \tau':(x,y)\to(a,b)}}^{k+1-d}c^{l(\tau')p(\tau')}=\frac{c}{|I(a)||I(b)|}\sum_{i=1}^{|I(a)|}\sum_{j=1}^{|I(b)|}\Bigg(\sum_{\substack{l(\tau')=1\\ \tau':(x,y)\to(I_i(a),I_j(b))}}^{k-d}c^{l(\tau')}p(\tau')+\sum_{d=1}^{k}t_d(I_i(a),I_j(b))\Bigg).$$

而 $\sum_{d=1}^{k}t_d(I_i(a),I_j(b))=\frac{1-c}{2}\sum_{\substack{l(\tau)=1\\ \tau:x\to y\cup\tau:y\to x}}^{k}c^{l(\tau)}p(\tau)$.

我们已假设第 k 步时公式成立:

$$M_k(I_i(a),I_j(b))=\sum_{\substack{l(\tau')=1\\ \tau':(x,y)\to(I_i(a),I_j(b))}}^{k-d} c^{l(\tau')p(\tau')}+\frac{1-c}{2}\sum_{\substack{l(\tau)=1\\ \tau:x\to y\cup\tau:y\to x}}^{k} c^{l(\tau)}p(\tau),$$

因此, $\sum_{\substack{l(\tau')=1\\ \tau':(x,y)\to(a,b)}}^{k+1-d} c^{l(\tau')p(\tau')}=\frac{c}{|I(a)||I(b)|}\sum_{i=1}^{|I(a)||I(b)|}\sum_{j=1}(M_k(I_i(a),I_j(b)))$ 成立.

由此得出 $M_{k+1}(a,b)=M'_{k+1}(a,b)$. 得证. □

张应龙(1979－),男,陕西绥德人,博士,讲师,CCF 会员,主要研究领域为图数据管理.

E-mail: zhang_yinglong@126.com

李翠平(1971－),女,博士,教授,博士生导师,CCF 杰出会员,主要研究领域为数据仓库和数据挖掘,信息网络分析,流数据管理.

E-mail: cuiping_li@263.net

陈红(1965－),女,博士,教授,博士生导师,CCF 高级会员,主要研究领域为数据仓库与数据挖掘,流数据分析与管理,传感器网络数据管理.

E-mail: chong@ruc.edu.cn