社交网络中社区领袖的挖掘算法

杨国如，范 磊
(上海交通大学信息安全工程学院，上海 200240)

[摘　要] 社交网络中用户和用户之间通过关注而产生联系形成社区。因此，文中借鉴 PageRank 算法，将传统上把影响力平均分配给关注的人的做法加以改进，依据用户间的亲密程度将影响力按比例分配给关注的人，从而生成新的 UserRank 算法。算法经过多次迭代计算后，社区中每个用户的影响力收敛并趋于稳定，影响力值最大的用户，就是社区领袖。实验表明，本算法能更快更有效地挖掘出社区领袖。

[关键词] 社交网络；社区领袖；PageRank；用户影响力
[中图分类号] TP393　　[文献标识码] A　　[文章编号] 1009-8054(2014)04-0122-05

Algorithm for Community Leader Detection in Social Network Services

YANG Guo-Ru，FAN Lei
(School of Information Security Engineering，Shanghai Jiaotong University，Shanghai 200240，China)

[Abstract] The users link to each other by following and form communities in social network. This article makes improvements based on the PageRank algorithm to generate a new algorithm UserRank for the detection of community leader. Traditional algorithm transfers the average of influence to the every user followed. The new algorithm transfers the influence in proportion to the every user followed. Influence of community users reaches to convergence and stable after algorithm iterations. The user with the highest-ranking influence is the community leader. Experimental results show that the new algorithm detects the community leaders out more quickly and more efficiently.

[Keywords] social networking service；community leader；PageRank；user influence

0　引言

研究社交网络的一个课题就是社区领袖的挖掘，领袖挖掘就是识别网络中最重要节点，这涉及到社区中心性分析，常用的标准有特征向量中心性 [1]。在社交网络中，用户影响力的大小体现了用户在网络中所处地位的重要程度。影响力大的用户在信息的交流和分享过程中越能施加影响 [1]。已经有很多文章对用户影响力做了研究，如 Meeyoung Cha[2] 等人利用 Twitter 上的用户数据通过 3 个方面来研究用户影响力：用户的粉丝数量 (Indegree)、用户的被转发次数 (Retweets) 和用户的被提及次数 (Mentions)。Daniel M. Romero[3] 等人提出衡量一个用户的影响力不仅要考虑粉丝的 Influence，还要考虑粉丝的 Passivity(忠诚度)。作者综合了 Influence 和 Passivity 后，提出一种算法用来判断一个用户的影响力。Jie Tang[4] 等人提出主题亲近传播 (Topical Affinity Propagation，TAP) 对大型网络中主题层次的社会影响进行建模。

文中观察到用于衡量网页重要性的 PageRank 算法 [5]，同样适用于衡量社区网络中用户的重要性。因此，通过借鉴 PageRank 算法并加以改进，生成新的 UserRank 算法，用来度量社区网络中用户的影响力大小。

收稿日期：2013-11-15
作者简介：杨国如，1978 年生，男，硕士，研究方向为数据挖掘；范磊，1975 年生，男，副教授，研究方向为网络安全管理等。

1 PageRank 算法介绍

PageRank，网页排名，是一种由搜索引擎根据网页之间相互的超链接计算网页等级的技术。作为网页排名的要素之一，PageRank 算法被广泛应用于度量网页重要性，它根据网页之间的链接结构来给每个网页打分，依据分数高低给出排名。算法把从 A 页面到 B 页面的链接解释为 A 页面给 B 页面的投票，根据投票的数量和质量来决定 B 页面的等级。投票的数量越多，意味着越多的页面认可 B 页面，则 B 页面的等级会越高；投票的质量越好，意味着越重要的网页认可 B 页面，则 B 页面的等级也会越高。Google 用它来体现网页的相关性和重要性，在搜索引擎优化操作中经常被用来评估网页优化的成效因素之一。PageRank 算法过程如下：

如图 1 所示，页面 B、C、D……链接到页面 A，则页面 A 的 PageRank 计算公式简单版本 [6] 为：

$$R(A)=\frac{R(B)}{E(B)}+\frac{R(C)}{E(C)}+\frac{R(D)}{E(D)}+\cdots=\sum_{V_i\in M}\frac{R(V_i)}{E(V_i)}$$

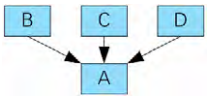

图 1 网页链接关系

考虑到归一化以及随机浏览等因素，简单版本的公式修改为完整版本 [6]：

$$R(A)=\frac{1-q}{N}+q\cdot\sum_{V_i\in M}\frac{R(V_i)}{E(V_i)}$$

公式中各变量含义如下：

V_i：网络中的网页。

$R(V_i)$：网页 V_i 的 PageRank 值。

$E(V_i)$：网页 V_i 链接至其他网页的链接数 (出链)。

M：链接至网页 A 的网页集合。

q：阻尼系数，取 0.85[6]。

N：网页总数。

所以，一个页面的 PageRank 是由其他页面的 PageRank 迭代计算得到的。如果给每个页面赋值一个随机的非零 PageRank 值，经过不断的迭代重复计算，这些页面的 PageRank 值会趋向于收敛的状态。然后对 PageRank 值进行由大到小排序，排名靠前的页面，其重要性就更高。

2 传统 PageRank 算法的不足

传统的 PageRank 算法通过利用网页之间的链接结构迭代计算出页面的 PageRank 值，再依据 PageRank 值的排名来判断网页的重要性，从而实现了将链接结构作为排名的因素。

在计算排名过程中，算法将同一页面的所有链出到的页面都同等对待。在迭代过程中，将自身的 PageRank 值平均分配给每个链出页面。公式中 $\frac{R(V_i)}{E(V_i)}$ 说明了页面 V_i 的 PageRank 值平均分配给它链出的每个页面。

事实上，同一页面的所有链出到的页面的重要性是有差别的，把 PageRank 值平均分配给所有的链出页面的做法，在一定程度上影响了网页的排序质量。文中认为这是传统 PageRank 算法一个值得改进的地方。

3 对 PageRank 算法的改进

网络上的网页之间是链接的关系，社区中的用户之间是关注的关系，这二者之间的结构是相似的。因此，引入 PageRank 算法并加以改进，用于计算社区中用户的影响力。为了将算法的应用拓展到社交网络中社区领袖的挖掘，文中把用户的影响力对应于网页的 PageRank 值，把用户的关注关系对应于网页的链接关系。

某个用户关注了特定的一些用户，这些被关注的用户之间并非同等重要，而是有所区别。迭代计算时从该用户处获取的影响力也应当有所区别而不是获取到平均值。

区别的方法就是加以考虑被用户关注的人和用户之间的亲密程度。亲密程度不同，在计算用户分配影响力时，获取到的用户分配的影响力就不同。如果被用户关注的人和用户共同关注了第三个人，则说明他们二位有相同的关系或者有相同的兴趣爱好。共同关注的数量越多，表明他们越亲密，分配到用户影响力的比例也应该越多。通俗的理解就是共同朋友越多，关系就越亲密，分配到的影响力就应当越多。

用户影响力的传递可以从两个方面进行理解：

1) 用户影响力的贡献。一个用户的影响力平均地 (或

者按比例地) 分配给该用户所关注的用户。

2) 用户影响力的获取。一个用户的影响力来源于该用户粉丝 (用户 A 关注了用户 B，则称 A 为 B 的粉丝) 的影响力所做出的贡献之和。

在数量上，用户的粉丝越多，贡献给该用户的影响力就越多；在质量上，粉丝的影响力越高，贡献给该用户的影响力也越多。假设有 5 位用户，他们之间的关注关系如图 2 所示。

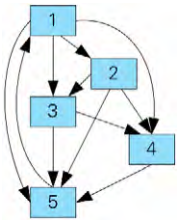

图 2 用户关注关系

从图 2 中观察到用户 1 关注了 4 位用户，传统的 PageRank 算法会把用户 1 的影响力平均分配给这 4 位用户。文中认为，这 4 位用户和用户 1 的关系亲密度不同，分配到的用户 1 的影响力也理应体现出不同。用户 2 关注了用户 3，4，5，说明用户 2 和用户 1 共同关注的人有 3 位。同理，用户 3 和用户 1 共同关注了 2 个人，用户 4 和用户 1 共同关注了一个人，用户 5 和用户 1 没有共同关注的人。和用户 1 共同关注的人越多，说明和用户 1 有更多共同的兴趣和关系，也就是关系更亲密，应该获得更多的用户 1 的影响力。可以通过计算共同关注的人数的比例来体现这种亲密程度。

总的共同关注人数为 3+2+1+0=6，由此，用户 2 分配到的比例为 3/6，同理，用户 3 分配到的比例为 2/6，用户 4 分配到的比例为 1/6，用户 5 分配到的比例为 0/6。定义 V_i 为社区中的节点，$F(V_i, V_j)$ 为节点 V_i 和节点 V_j 共同关注的用户数，W 为 V_a 关注对象的集合，则节点 V_a 影响力分配比例的公式为：

$$\frac{F(V_i, V_a)}{\sum_{V_j \in W} F(V_j, V_a)}$$

考虑到所有节点共同关注的人数全为 0 的情况，计算比例的时候，把被用户 1 关注的人本身也计算在内，分配比例的公式修改为：

$$\frac{F(V_i, V_a)+1}{\sum_{V_j \in W} (F(V_j, V_a)+1)}$$

这样，例子中总的共同关注人数为 (3+1)+(2+1)+(1+1)+(0+1)=10，用户 2 分配的比例为 (3+1)/10，用户 3 分配的比例为 (2+1)/10，用户 4 分配的比例为 (1+1)/10，用户 5 分配的比例为 (0+1)/10。

由公式可以观察到，当用户 1 的所有关注的人和用户 1 本身之间都没有共同关注的人的时候，影响力分配比例值就等于平均分配。平均分配只是按比例分配的一种特殊情形。本公式体现了按比例分配是平均分配的扩展情形。影响力分配的比例确定后，在每次迭代过程中该比例都保持不变。

考虑到归一化等因素，文中提出 UserRank 算法，即用户获取的影响力的公式完整版：

$$R(V_a)=\frac{1-q}{N}+q\cdot\sum_{V_i \in M}\left(\frac{F(V_i, V_a)+1}{\sum_{V_j \in W}(F(V_j, V_a)+1)}\cdot R(V_i)\right)$$

公式中各变量含义如下：

V_i：社区中的节点。

$R(V_i)$：节点 V_i 的影响力值 (类似于 PageRank 中的 Rank 值)。

$F(V_i, V_a)$：V_i 与 V_a 共同关注的人数。

M：节点 V_a 的所有粉丝集合。

W：V_a 关注对象的集合。

q：阻尼系数，取 0.85[6]。

N：社区总节点数。

4 实验结果

实验中，利用新浪官方提供的 API 从新浪微博的服务器上下载微博的用户数据，并从数据中用标签算法挖掘出社区，从中选取出 5 个社区数据进行分析。这 5 个社区数据的用户数分别为 181 个、267 个、814 个、1 824 个和 9 300 个，这样在用户数量的规模上具有一定的代表性。

用 UserRank 算法对社区中的用户数据进行计算以获取每个用户的影响力，并记录用户影响力的变化过程。当每个用户前后 2 次迭代获得的影响力差值小于 0.001 的时候，认为实验结果达到了收敛状态，用户影响力趋于稳定。将最后一次迭代获得的已趋于稳定状态的影响力值降序排列进行观察，取出最大值，

其对应的用户就是该社区的社区领袖。

Kollias G[7] 等人研究发现，在对 PageRank 进行计算时，采用异步迭代的方法比采用同步迭代的方法，能更快地达到收敛状态。该实验计算社区用户影响力时，也借鉴此种做法。迭代计算用户影响力的时候，如果粉丝的影响力值使用前一次迭代所获得的值，定义此种做法为同步法。粉丝的影响力值使用最近一次迭代所获得的值，定义此种做法为异步法。用 PageRank 算法对社区数据进行计算，获取社区领袖，然后比较两种算法的计算结果，如表 1 所示，表中社区领袖用 UserID 表示。

表 1 同步迭代社区领袖

	PageRank	UserRank	结果
实验 1	1162178432	1162178432	一致
实验 2	2749723927 2770967391	2749723927 2770967391	一致
实验 3	2459524747	2459524747	一致
实验 4	2866942642	2866942642	一致
实验 5	2847571251	2847536195 2847536361	不一致

表 1 记录的数据是采用同步迭代法时，两种算法计算出的社区领袖，从中观察到，实验 5 中，二者的结果不一致。这是因为 PageRank 算法计算出的用户影响力的第一名是用户 2847571251，第二名是用户 2847536195 和用户 2847536361；而 UserRank 算法计算出的第一名是用户 2847536195 和用户 2847536361，第二名是 2847571251。即两种算法计算出的第一名和第二名互相换了位置。

从实验数据中观察到用户 2847571251 的粉丝有 99 个，其中，有 26 个粉丝用 UserRank 算法计算出的粉丝分配给用户的影响力比例权重比用 PageRank 算法计算出的大，但有 73 个用户是变小或相等的。

而用户 2847536195 和用户 2847536361 的粉丝都是 158 个，且这些粉丝用 UserRank 算法计算出的粉丝分配给用户的影响力比例权重都比用 PageRank 计算出的大。这说明用 UserRank 算法，用户 2847536195 和用户 2847536361 获取的影响力变多了，而用户 2847571251 获取的影响力变少了。综合起来，用户 2847536195 和用户 2847536361 由并列排名第二变成了并列排名第一。

在另外 4 个实验中，用两种算法挖掘社区领袖所得到的结果是一样的。其中，实验 2 中，两种算法计算出的领袖都有 2 个，他们的影响力并列第一。同步迭代的实验很好地验证了新算法的挖掘准确度。

表 2 记录的数据是采用异步迭代法时，两种算法计算出的社区领袖。从中观察到，实验 2 中，用户 2749723927 和用户 2770967391 的影响力用 PageRank 算法计算的结果相等，并列第一；用 UserRank 算法计算的结果是用户 2749723927 排名第一，而在 PageRank 算法中并列第一的用户 2770967391 在这里却是排名第二。

表 2 异步迭代社区领袖

	PageRank	UserRank	结果
实验 1	1162178432	1162178432	一致
实验 2	2749723927 2770967391	2749723927	不一致
实验 3	1814766447	1814766447	一致
实验 4	2866942642	2866942642	一致
实验 5	2847571251	2847571251	一致

在另外 4 个实验中，用两种算法挖掘社区领袖所得到的结果一样。异步迭代的实验很好地验证了新算法的挖掘准确度，而且精确度也更高，能把并列第一的用户也进一步区分出来。表 3 是同步迭代法下，两种算法达到收敛时所需的迭代次数。

表 3 同步迭代次数

	用户数	PageRank	UserRank	提高效率 /%
实验 1	1 824	15	15	0
实验 2	267	37	41	-10
实验 3	814	36	35	2
实验 4	9 300	114+	44	>61
实验 5	181	35	39	-11

表 4 是异步迭代法下，两种算法达到收敛时所需的迭代次数。从表 4 观察到，算法在结果达到收敛并统计所需的迭代次数时发现，社区中用户数量较少的时候，在同步迭代法和异步迭代法下，UserRank 算法

与 PageRank 算法相比，在迭代次数上并没有明显优势甚至略差，但随着社区中用户数量的增加，在迭代次数上的优势明显突出。实验 4 中，用 PageRank 算法迭代了 114 轮后，仍然没有达到收敛，而 UserRank 分别只要 44 次和 27 次就达到了收敛，效率提高超过 61% 和 76% 的幅度。

表 4 异步迭代次数

	用户数	PageRank	UserRank	提高效率 /%
实验 1	1 824	9	9	0
实验 2	267	25	25	0
实验 3	814	24	25	-4
实验 4	9 300	114+	27	>76
实验 5	181	21	23	-9

表 5 是统计 UserRank 算法在计算用户影响力的时候，算法在采用同步法和异步法并达到收敛时分别用到的计算迭代次数。

表 5 迭代次数

	同步迭代	异步迭代	提高效率 /%
实验 1	15	9	40
实验 2	41	25	39
实验 3	35	25	29
实验 4	44	27	39
实验 5	39	23	41

从表 5 中观察到，算法在结果达到收敛并统计所需的迭代次数时发现，异步迭代法比同步迭代法所需的迭代次数明显少很多，5 个实验的结果都验证了这一点。异步迭代法比同步迭代法提高了 29%~41% 的效率。

5 结语

文中通过借鉴 PageRank 算法，并在传统算法的基础上提出了改进，将传统算法把用户影响力平均分配给关注的人的做法，改进为将用户影响力按用户和关注的人之间的亲密程度不同以不同的比例来分配。算法复杂度比传统算法有一些增加。

实验中使用新浪微博的用户数据，通过同步迭代计算和异步迭代计算的实验均验证了 UserRank 算法挖掘社区领袖的准确度；如期验证了新算法中的共同好友数所起的作用；在采用异步迭代计算时，新算法精度更高；实验结果也表明了新算法采用异步迭代计算比采用同步迭代计算效率有明显的提高。

通过分析可推断，当社区中用户之间的联系比较少、比较稀疏时，改进后的算法和传统算法相比，数据处理结果较为相近；当社区中用户之间的联系比较多、比较复杂时，改进后的算法和传统算法相比，明显有优势。这是因为传统算法只是改进后算法的一个特殊情形，改进后的算法是传统算法的扩展。改进后的算法比传统算法能更快更有效地挖掘出社区的领袖。

参考文献

[1] TANG L，LIU H. Community Detection and Mining in Social Media[J]. Synthesis Lectures on Data Mining and Knowledge Discovery，2010，2(1)：1-137.

[2] CHA M，HADDADI H，BENEVENUTO F，et al. Measuring User Influence in Twitter：The Million Follower Fallacy[J]. ICWSM，2010，10：10-17.

[3] ROMERO D M，GALUBA W，ASUR S，et al. Influence and Passivity in Social Media[M]//Machine Learning and Knowledge Discovery in Databases. Springer Berlin Heidelberg，2011：18-33.

[4] TANG J，SUN J，WANG C，et al. Social Influence Analysis in Large-scale Networks[C]//Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM，2009：807-816.

[5] PAGE L，BRIN S，MOTWANI R，et al. The PageRank Citation Ranking：Bringing Order to the Web[R]. Technical report，Stanford Univ.，1999.

[6] BRIN S，PAGE L. The Anatomy of a Large-scale Hypertextual Web Search Engine[J]. Computer Networks and ISDN Systems，1998，30(1)：107-117.

[7] KOLLIAS G，GALLOPOULOS E，SZYLD D B. Asynchronous Iterative Computations With Web Information Retrieval Structures：The PageRank Case[R]. arXiv Preprint cs/0606047，2006.