基于社区的关键节点挖掘算法①

陆晓野，陈　玮

(上海理工大学 光电信息与计算机工程学院，上海 200093)

摘　要：在对大型网络进行关键节点挖掘方面，传统方法效率低下。针对这一缺陷，提出了一种基于社区的关键节点挖掘算法，首先对社区发现算法进行改进，然后提出基于节点频度中心度的挖掘算法。实验结果表明，新算法对社区进行关键节点挖掘时，不仅挖掘的影响度得到保证，而且效率显著提高。

关键词：关键节点；社区发现；社会网络；频度中心度；影响度

Key-Nodes Mining Algorithm Based on Communities

LU Xiao-Ye, CHEN Wei

(School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China)

Abstract: When mining the key-nodes from large-scale networks, the traditional methods are poor efficiency. To address this defect, a new algorithm, which based on communities, is presented for mining the key-nodes. First, it improved the community detection algorithms, and then put forward an algorithm based on degree centrality of the node for mining the key-nodes. The experimental results show that when applying the new algorithm to mine the key-nodes from communities, not only the influence degree of mining is guaranteed, but also the efficiency is improved significantly.

Key words: key-nodes; community detection; social networks; degree centrality; influence degree

现实生活中，社会网络无处不在，如人际关系网、Facebook、Myspace、论文合著网、博客等。社会网络由节点和连接节点之间的边组成，如在论文合著网中，节点代表论文作者，边代表作者之间的合作关系。近年来，在社会网络中挖掘关键节点（或挖掘具有影响力的节点）已成为该领域一个比较热门的研究课题。SIGKDD 国际会议已经连续四年在社会网络分析和挖掘方面举行研讨会。

在社会网络中进行关键节点挖掘有着广泛的应用。在计算机网络安全防御方面，不用盲目地花大成本生产安全软件，并对整个网络进行防御，而是先挖掘出那些容易传播病毒的关键计算机，再集中注意力对它们进行防御，达到低成本高效率的目的。在市场营销方面，挖掘出具有影响力的潜在客户，对他们进行新产品的免费试用，进而对其周围亲戚朋友起到积极的影响作用，影响他们也来购买新产品，他们又会影响他们自己的朋友，口口相传，实现“病毒式营销”。

在社会网络中，挖掘 k 个最具有影响力节点的最优问题是一个 NP-问题[1]。目前的相关文献，采用贪婪算法来解决该问题，影响度都能得到保证，但挖掘大型网络时，效率低下，算法运行时间让人难以接受。针对这一问题，本文提出了一种新的算法——基于社区的关键节点挖掘算法。该算法的思想来源于社会网络普遍存在的特性——社区结构[2,3]，社区内部节点联系紧密，而社区之间联系稀疏。

1　相关研究

在关键节点挖掘方面，已有大量相关研究。Gruhl 等人[4]从宏观特征上，挖掘博客圈具有影响力的博客，研究博客圈的主题如何传播，以及它们如何维持有效性。Lappas 等人[5]提出 k-Effectors 问题，即在给定的传播模型下，挖掘 k 个活跃节点，并使这 k 个活跃节

① 收稿时间:2011-06-07;收到修改稿时间:2011-07-16

点最能表达当前网络的激活状态。Chen 等人[6]提出 NewGreedy 算法和 MixedGreedy 算法，MixedGreedy 算法是 NewGreedy 算法的改进算法，MixedGreedy 算法的第一轮采用 NewGreedy 算法，之后使用 CELF 算法，经验证，MixedGreedy 的性能确实得到了提高

社区发现也是社会网络分析领域的一个热点课题。早期的算法有 K-L 算法、谱平分法、最大流最小分割法等，然而，这些算法都需要预先知道社区的个数或社区规模的大小，在真实社会网络中，这是不可能的，因此这些方法的应用很有局限性。Newman 等人[3]提出了基于边介数的 GN 算法，该算法自顶向下对网络进行分解，当模块度 Q 取最大值时算法终止，

$$Q=\sum_i\left(e_{ii}-a_i^2\right)=Tre-\left\|e^2\right\| \quad (1)$$

GN 算法要求计算所有边的介数，而每次去掉具有最大介数值的边，都要重新计算所有边的介数，不断重复，计算量非常大，不适用于大型网络，时间复杂度为 $o\left(n^3\right)$。Newman 等人[7]又提出基于模块度增量的快速算法，该算法自底向上对网络节点进行合并，时间复杂度为 $o(md\log n)$，其中，n 为节点数，m 为边数，d 为聚类图的深度，在稀疏图中，$d\approx\log n$，$m\approx n$，则时间复杂度为 $o\left(n\log^2 n\right)$，可在线性时间内完成稀疏图的分析，该算法适于挖掘大型网络，广被接受。

2 基于社区的关键节点挖掘算法

2.1 基本思想与符号定义

社会网络用图来表示，节点代表网络中的个体，边代表个体之间的联系，本文只考虑无向无权重图。为方便后文算法描述，在表 1 中，给出基本的符号定义，并结合表 1 的符号定义，提出本文的问题定义。

表 1 符号定义

符号	描述
G=(V, E)	图 G 中，节点集合 V，边集合 E
C_i	第 i 个社区
Q	模块度
k_i	节点 i 的度
T	关键节点集合
d	路径距离参数
$N_d(t)$	距离节点 t 为 d 范围内的所有邻近节点
| |	基数，即集合的元素个数

问题定义：给定一个无向无权重图 G(V, E)，节点数 n 为|V|，边数 m 为|E|，挖掘 k 个关键节点，使得从这 k 个关键节点出发的消息传播最大化。其中，消息传播模型的限制条件为：如果节点 i 能够影响（或激活）节点 j，那么节点 i 在距离 d 内必须可达节点 j。

基于社会网络的社区结构特性，把上面问题分解为两个步骤：1）社区发现，对 Newman 等人[7]提出的快速算法进行改进；2）基于节点频度中心度，对各个社区进行关键节点挖掘。算法主要步骤，如图 1 所示。

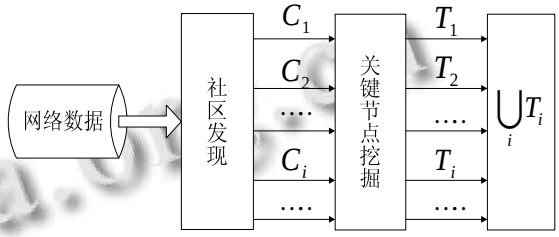

图 1 基于社区的关键节点挖掘算法的主要步骤

2.2 社区发现

采用 Newman 等人[7]提出的快速算法进行改进，加入社区规模门限值 g，出于如下原因：①Newman 等人提出的快速算法时间复杂度为 $o(md\log n)$，效率高，能适用于分析大型网络；②文献[7]，挖掘出的前十个最大社区包含整个网络 87%的节点，若不设置社区规模门限值 g，则对其挖掘关键节点效率低下，与直接基于原始网络进行挖掘没有区别；③大社区隐含许多语义丰富的小社区，而小社区内部节点性质相似，联系紧密[8]，因此，设置社区规模门限值 g，把大社区进一步细分，相当于把一个比较广泛的领域细分为许多更具体的群体，有利于关键节点的挖掘。

根据 Newman 等人[7]提出的快速算法，需要定义如下数据结构：

a) 矩阵 ΔQ_{ij}：社区 C_i 和社区 C_j 合并后的模块度；

b) 堆 H：保存矩阵 ΔQ_{ij} 中各行的最大元素；

c) a_i：$a_i=\sum_j\Delta Q_{ij}$，用一个向量容器来保存 a_i。

该算法是自底向上的聚类方法，起初把每一个节点看作一个单独的社区，且设置初始值公式如下：

$$\Delta Q_{ij}=\begin{cases}1/2m-k_ik_j/(2m)^2 & \text{如果 i,j 相连}\\ 0 & \text{否则}\end{cases} \quad (2)$$

$$a_i=k_i/(2m) \quad (3)$$

加入社区规模门限值 g 后，得到的改进算法步骤如下：

(a) 根据公式（2）、（3），计算 ΔQ_{ij} 和 a_i，且把矩阵 ΔQ_{ij} 中每行的最大元素保存到堆 H 中；

(b) 找出堆 H 中最大的元素 ΔQ_{ij}；若 $\Delta Q_{ij}>0$，则转到(c)；否则，转到(d)；

(c) 合并社区 C_i 和社区 C_j，更新矩阵 ΔQ_{ij}（由公式（4）~（6））、堆 H、a_i、$Q=Q+\Delta Q_{ij}$；转到(b)；

(d) 如果挖掘得到的全部社区，它们的规模（即包含的节点数）都不超过给定的门限值 g，则算法结束。否则，把那些规模大于门限值 g 的社区继续进行社区挖掘，针对每一个这样的社区，转到(a)继续执行。

步骤(c)，合并社区 C_i 和社区 C_j 时，把矩阵 ΔQ_{ij} 的第 i 行和第 i 列删除，同时标记第 j 行和第 j 列，且第 j 行和第 j 列更新的规则如下：

a) 如果社区 C_i 和社区 C_j 都与社区 C_k 相连，那么

$$\Delta Q'_{jk}=\Delta Q_{ik}+\Delta Q_{jk} \tag{4}$$

b) 如果社区 C_i 与社区 C_k 相连，而社区 C_j 与社区 C_k 不相连，那么

$$\Delta Q'_{jk}=\Delta Q_{ik}-2a_j a_k \tag{5}$$

c) 如果社区 C_j 与社区 C_k 相连，而社区 C_i 与社区 C_k 不相连，那么

$$\Delta Q'_{jk}=\Delta Q_{jk}-2a_i a_k \tag{6}$$

2.3 关键节点挖掘

社区发现之后，分别对各个社区进行关键节点挖掘，然后把各个社区的关键节点进行整合，$T=\bigcup_i T_i$。

针对每一个社区挖掘关键节点，本文提出基于节点频度中心度的挖掘算法。根据第 2.1 节的问题定义，要挖掘 k 个关键节点，则关键节点数占总节点数的百分比为 $p=k/n$，那么对社区 C_i 要挖掘的关键节点数为

$$h_i=\lceil |C_i|\times p \rceil \tag{7}$$

对 h_i 向上取整。

基于节点频度中心度挖掘算法的具体步骤如下：

(a) 初始化：社区 C_i 的关键节点集合 $T_i=\phi$，关键节点数 h_i（由公式（7）得出）；

(b)在社区 C_i 中，选取未做标记且具有度最大的节点 t 作为社区 C_i 的关键节点，则 $T_i=T_i\cup\{t\}$；如果同时存在几个度最大的节点，则随机选择其中一个；

(c)在距离关键节点 t 为 d 的范围内，查找关键节点 t 的所有邻近节点 $N_d(t)$；

(d) 在社区 C_i 中，给关键节点 t 及其所有邻近节点 $N_d(t)$ 做上标记；

(e)社区 C_i，若已没有未标记的节点，或所得关键节点数 $|T_i|\geq h_i$，则算法停止；否则转到(b)继续执行。

对本文提出的基于节点频度中心度的挖掘算法进行总结，得出以下几点：①参数 d 和参数 k 是反向的，距离 d 越大，则表明邻近范围越广，能覆盖的邻近节点越多，整个社区被覆盖得越快，那么 k 可适当减小；反之，d 越小，则 k 适当增大，可不断调整 d 和 k，优化挖掘效果；②文献[9]的实验结果表明，一个小社区代表一个具有共同话题或共同兴趣爱好的群体，可见小社区内部节点联系紧密，节点平均度高，因此采用基于节点频度中心度的挖掘算法是可行的；③算法步骤简单，且挖掘出的 k 个关键节点，分别取自各个社区，具有代表性，全局性，能覆盖整个网络。

3 实验

3.1 实验坏境与数据集

实验环境：①硬件：Intel Pentium Dual 2.16GHz 1GB；②系统环境：Windows XP；③开发环境：VC++6.0。

数据集：包括三个数据集，都是来源于网址[10]。1) Zachary's karate club：有 34 个节点，78 条边，节点代表俱乐部成员，边代表成员间的关系；2) American College Football：有 115 个节点，613 条边，节点代表足球队，边代表两个球队之间的比赛；3) Coauthorships in network science：有 1461 个节点，2742 条边，节点代表科学文献作者，边代表两个作者之间有过合作。

数据预处理：网址[10]的数据文件格式为.gml，包括节点信息和边信息，如：数据结构为 node [id 1]和 edge [source 28　target 3]，边由源节点和目标节点组成。实验数据仅需要边信息，因此从源数据文件中删除前面的节点信息，再运行程序 transformData.cpp，从.gml 文件提取边信息，转换并保存到.txt 文件，转换过程为 edge [source 28　target 3]→(28　3)和(3　28)，数据结构为每条边由两个节点编号组成，且各占一行。

3.2 算法实现

整个挖掘算法分为两个主要阶段，①社区发现，采用文献[7]的快速算法进行改进，该算法的源代码可

从网址[11]下载，在该源代码中加入社区规模门限值 g 即可，本实验不对社区发现阶段深入研究。②基于节点频度中心度的关键节点挖掘算法，步骤在第 2.3 节中描述，下面给出该算法的伪代码。

输入: C_i, int d, int h_i; 输出: set<int> T_i;

Begin: set<int> $T_i = \Phi$;

map<int, set<int> > community_i $\leftarrow C_i$;

{ int t ← select_max_degree(community_i);

$T_i = T_i \bigcup \{t\}$;

set<int> neighbour_t ← find_neighbour(t,d);

neighbour_t = neighbour_t∪{t};

delete_neighbour(community_i, neighbour_t);

}while(community_i $\neq \Phi$ && $|T_i| < h_i$);

output T_i; End;

算法要求输入一个社区网络数据，距离参数 d 和关键节点数 h，则输出关键节点集合 T。

3.3 实验结果与分析

假设第 3.1 节的每一个数据集是一个单独的社区，测试本文实现的基于节点频度中心度的挖掘算法。实验结果如表 2、3、4、5 所示，对于每一个数据集，改变传播距离参数 d，得到不同的运行结果，k 为关键节点数，p 为关键节点数与总节点数的百分比，c 为影响度或覆盖率，即接收到信息的节点数与总节点数的百分比，T 为关键节点集合，time 为运行时间，单位秒。

表 2 数据集 Zachary's Karate Club (c≥0.90)

d	1	2	3	4
k	2	2	1	1
p	5.88%	5.88%	2.94%	2.94%
c	91.18%	85.29%	97.06%	100%
T	{1,34}	{6,34}	{34}	{34}
time	0	0	0	0

表 3 数据集 American College Football (c≥0.90)

d	1	2	3	4
k	11	3	1	1
p	9.57%	2.61%	0.87%	0.87%
c	92.17%	93.04%	100%	100%
time	0.016	0.016	0.015	0

表 4 数据集 Coauthorships (c≥0.90)

d	2	3	5	7	10
k	235	231	218	200	195
p	16.08%	15.81%	14.92%	13.69%	13.35%
c	90.01%	90.08%	90.01%	90.08%	90.01%
time	1.578	1.5	1.375	1.25	1.25

表 5 数据集 Coauthorships (c≥0.80)

d	2	3	5	7	10
k	172	168	157	141	137
p	11.77%	11.50%	10.75%	9.65%	9.38%
c	80.01%	80.15%	80.08%	80.01%	80.08%
time	1.468	1.391	1.281	1.156	1.11

算法的终止条件为影响度 c 的取值，表 2~ 4 的终止条件为 c≥90%，而表 5 的终止条件为 c≥80%。如表 2～表 5 所示，在影响度 c 保持不变时，信息传播距离参数 d 与关键节点数 k 的取值是反向的，d 越大，则 k 越小。市场营销，要求新产品达到一定的影响度 c，并根据市场信息传播模型计算出 d，由本文算法可确定 k 个有影响力的潜在客户，进行有针对性的营销。

影响度 c 用于衡量信息在社会网络中得到传播的程度，如广告投入得到的影响度，如表 2~表 4 所示，影响度 c≥90%，保持很高的水平。在 d 和 p 取较小值时，也能达到很高的影响度，如表 3 中，d=2，p=2.61%，c=93.04%；d=3，p=0.87%，c=100%，体现社会网络的社区结构特性，社区内部节点联系紧密，节点之间连接的平均路径短，如朋友圈中，联系紧密，挖掘出几个有影响力的人进行新产品试用，也能达到很好的效应。给定距离参数 d，且在一定数量 k 下，本文算法可计算出大概的影响度 c，如表 4 和表 5 所示，影响度 c 由 90%变为 80%，对应 d 相等时，k 明显变小，表明用于营销的资源减少，得到的营销效果就会降低。

三个数据集的节点数分别为 34、115、1461，节点数呈指数增加，但如表 2~表 5 所示，运行时间 time 没有显著增大，均低于 1.6 秒，算法效率高，具有快速收敛性。假设社会网络包含 40 万个节点，设 g=1500，且社区发现得到的社区平均规模为 1000 个节点，则大概有 400 个社区，对每个社区挖掘关键节点需 1.5 秒，那么总的关键节点挖掘时间为 600 秒（不包括社区发现时间）。由此，本文算法适用于挖掘大型网络。

(下转第 197 页)

从图 4 中的(a)、(b)、(c)三幅图可以看出，图(a)和图(b)中在相同迭代次数 20 代中，PSO-BP 的最好适应值随迭代次数的变化而变化，未达到稳定状态。而 HJPSO-BP 经过 3 代达到稳定状态。图(c)中 PSO-BP 在经过 50 代达到稳定。充分体现了 HJPSO-BP 比 PSO-BP 稳定性的提高，应用于齿轮热处理中提高了预测的精度和稳定性。

4 结语

文中建立HJPSO优化的BP神经网络模型，在PSO算法中引入Hooke-Jeeves模式搜索法提高了PSO算法的局部搜索能力及精度。用粒子群算法代替 BP 神经网络中的梯度下降法训练神经网络的连接权值和阈值，来改善 BP 算法的性能，使其不易陷入局部最小的缺点。此算法既发挥了 BP 神经网络在预测领域的优点，同时又结合了 PSO 算法全局搜索能力强、收敛速度快的特点。通过对齿轮热处理进行预测，结果表明利用改进的 BP 神经网络可以很好的实现输入到输出的映射，该方法有效地提高了神经网络的预测精度，为齿轮热处理质量的预测提供了指导作用。

参考文献

1 陈晖,周细应.汽车齿轮热处理工艺的研究进展.上海工程技术大学学报,2010,24(7):93–96.

2 李新城,等.齿轮选材及热处理工艺智能专家系统.金属学报,2004,40(10):1051–1054.

3 Dennis O, Charles M. Neural network forecasts of Canadian stock returns using accounting ratios. International Journal of Forecasting,2003,19(3):453–465.

4 俞欢军,等.混合粒子群算法研究.信息与控制, 2005,34(4): 500–504.

5 Eberhart RC, Shi Y. Particle swarm optimization: Developments, applications, and resources. Proc. of the 2001 Congress on Evolutionary Computation, 2001. 81–86.

6 Lv WZ, Fan HY, Leung AYT, Wong JCK. Analysis of pollutant levels in central HongKong applying neural network method with particle swarm optimization. Environmental Monitoring and Assessment, 2002: 217–230.

7 Yang DH. Run off prediction by BP networks model based on PSO. Journal of Hydroelectric Engineering, 2006,25:20–25.

8 董长虹.matlab 神经网络与应用.北京.国防工业出版社, 2005.64–104.

9 严武元,王少梅.PSO-BP 混合预测模型及在港口集装箱吞吐量预测中的应用.武汉理工大学学报,2007,31(3):525–528.

10 江思珉,等.水文地质参数反演的 Hooke-Jeeves 粒子群混合算法.同济大学,2010,21(5):606–611.

(上接第 253 页)

4 结语

根据社会网络的社区结构特性，提出基于社区的关键节点挖掘算法，而不是直接对整个网络进行关键节点挖掘。实验结果表明，本文算法适用于挖掘大型网络，且具有高影响度、高效率、快速收敛性。在社会网络规模成指数级增长的今天，本文算法具有实际意义和应用价值。

参考文献

1 Kempel D, Kleinberg J, Tardos E. Maximizing the spread of influence through a social network. ACM SIGKDD, 2003, 137–146.

2 Girvan M, Newman MEJ. Community structure in social and biological networks. Proc. Natl. Acad. Sci. USA, 2002, 99(12): 8271–8275.

3 Newman MEJ, Girvan M. Finding and evaluating community structure in networks. Phys. Rev. E, 2004.

4 Gruhl D, Guha R, Tomkins A. Information diffusion through blogspace. ACM SIGKDD, 2004,6(2):43–52.

5 Lappas T, Terzi E, Gunopulos D, et al. Finding Effectors in Social Networks. ACM SIGKDD, 2010.

6 Chen W, Wang YJ, Yang SY. Efficient influence maximization in social networks. ACM SIGKDD, 2009: 199–208.

7 Clauset AM, Newman EJ. Finding community structure in very large networks. Phys. Rev. E, 2004,70(6).

8 吴文涛,肖仰华,何震瀛,等.基于权重信息挖掘社会网络中的隐含社团.计算机研究与发展,2009,46:540–546.

9 吴龙庭,戴汝为,崔霞.一种局部最优社区挖掘方法.计算机应用研究,2009,26(8).

10 www–personal.umich.edu/~mejn/netdata/.

11 www.cs.unm.edu/–aaron/research/fastmodularity.htm.