计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2020,29(10):148−157 [doi: 10.15888/j.cnki.csa.007643]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于标签传播的拓扑势社区检测算法①

费　蓉[1], 李莎莎[1], 胡　博[2], 唐　瑜[1], 方金正[1]

[1](西安理工大学 计算机科学与工程学院, 西安 710048)
[2](北京华电优控科技有限公司, 北京 100193)
通讯作者: 费　蓉, E-mail: annyfei@xaut.edu.cn

摘　要: 基于拓扑势的社区检测通过节点的链接信息构造拓扑势域, 在拓扑势域内进行社区划分. 但实际划分过程存在大量孤立性社区. 带节点属性信息的社区检测问题作为社区的重要组成, 已成为社区检测的主要研究方向. 本文提出了一种结合标签传播的拓扑势社区检测算法 (TPCDLP). 首先, 结合标签传播思想将属性信息转换为节点间的链接权值. 其次, 把链接权值加入到拓扑势中构造拓扑势域. 再利用核心节点进行子群社区的划分. 最后, 利用子群社区间核心节点的距离进行社区划分. 在 3 个含标签属性的数据集上, 与 6 种算法对比, 该算法在改进的模块度 Q_{ov}^{E}、信息熵*Entropy*、社区重叠度*Overlap*和综合指标 F 上表现更优. 在 3 个真实社区上应用了该算法, 并与 3 种算法对比, 实验结果显示该算法在标准化互信息指标*NMI*上表现良好, 能够有效应用于实际问题.

关键词: 拓扑势; 标签传播; 社区检测; 数据场

引用格式: 费蓉,李莎莎,胡博,唐瑜,方金正.基于标签传播的拓扑势社区检测算法.计算机系统应用,2020,29(10):148–157. http://www.c-s-a.org.cn/1003-3254/7643.html

Topological Potential Community Discovery Algorithm Based on Label Propagation

FEI Rong[1], LI Sha-Sha[1], HU Bo[2], TANG Yu[1], FANG Jin-Zheng[1]

[1](Faculty of Computer Science and Engineering, Xi'an University of Technology, Xi'an 710048, China)
[2](Beijing Huadian Ucontrol Technology Co. Ltd., Beijing 100193, China)

Abstract: Community detection based on the topological potential constructs the topological potential field by the link information of nodes, in which the community can be partitioned. However, there are a large number of isolated communities in the actual division process. The problem of community discovery with node attribute information, as an important part of the community, has become the main research direction of community discovery. This paper proposes a topological potential community discovery algorithm combined with label propagation (TPCDLP). First, combining the thought of label propagation, the attribute information is converted into the link weights between nodes. Second, the link weights are added to the topological potential to construct the topological potential field. Then, the subgroup communities are partitioned by the core node. Finally, the communities are partitioned by using the distance of the core nodes between the subgroup communities. Compared with six algorithms on three datasets with label attributes, the TPCDLP performs better on the improved modular degree Q_{ov}^{E}, information entropy *Entropy*, community overlap degree *Overlap* and comprehensive index F.

Key words: topological potential; label propagation; community discovery; data field

① 基金项目: 国家自然科学基金 (61773313); 陕西省重点研发计划 (2017ZDXM-GY-098); 陕西省自然科学基础研究计划 (2020JM-709)
Foundation item: National Social Science Foundation of China (61773313); Key Research and Development Program of Shaanxi Province (2017ZDXM-GY-098); Fundamental Research Program of Natural Science of Shaanxi Province (2020JM-709)
收稿时间: 2020-03-17; 修改时间: 2020-04-14; 采用时间: 2020-04-24; csa 在线出版时间: 2020-09-30

众多复杂系统都可抽象成为网络模型, 如计算机网络、信息网络、社会网络和生物网络等得到了广泛应用[1], 社区检测问题对于研究复杂网络以及人类生活具有重要意义. 社区检测期望将链接最紧密的节点划分至同一社区, 有助于更好地了解整个社交网络, 进而有效利用资源[2]. 现实中, Facebook 等以朋友关系为基础的社交网络上, 通过社区检测可进行朋友推荐[3,4]. 另外也可以用社区检测对具有链接关系并且同兴趣的用户进行兴趣推送[5]. 除此之外, 还可用于交通网络中分析交通对城市功能社区 (商业区、居民区、学校等) 分布之间的关系[6].

近年来, 社区检测问题常归于以下类型: 基于图分割的社区检测, 需要提前定义分割社区个数及体积, 通过最小化社区间的链接边的数量实现社区划分, 如 Kemighan-Lin 算法和谱划分算法; 基于聚类的社区检测则是通过节点间的关系利用聚类的思想将其进行社区检测, 以 GN 算法[7]、Newman 贪心算法和 k-means 算法为代表; 基于模块度最大化的社区检测如 Louvain 算法, 利用模块度获取最优的网络社区划分; 基于非负矩阵的社区检测, 利用非负矩阵的思想将节点的链接矩阵进行分解得到节点社区归属矩阵, 如 LANMF 算法[8]; 基于标签的社区检测算法, 以 LPA 算法、CORP 算法和 LPPB 算法等为代表, 对每个节点随机生成标签, 逐轮刷新所有节点的标签, 直到所有节点的标签不再发生变化为止.

节点拓扑势的概念源于认知物理学中的数据场理论[9], 2009 年, 淦文燕提出了一种基于拓扑势的社区检测方法, 利用节点的链接信息构造拓扑势场, 在拓扑势场内进行社区划分[10]. 拓扑势原理近年来得到了长足的发展. 2018 年, Wang 在山谷结构的拓扑势场下基于节点位置进行分析, 设计 DOCET 算法[11]. 但拓扑势社区算法在实践中存在一种现象, 当获得的模块度值较高时, 社区的划分数量过大, 当社区网络过于复杂时, 真实数据集出现了很多孤立性节点或孤立性小社区. 基于拓扑势原理进行社区划分, 存在大量 3-4 节点孤立为一个社区的现象出现. 这种孤立社区的出现为现实的推送, 社区的扩大等带来影响. 近期研究显示, 社区划分不再单纯的考虑链接结构, 而是通过增加节点的属性信息进行社区划分. 节点的属性信息越来越受到关注[12].

本文面向含标签属性的社区检测问题, 针对上述基于拓扑势进行的社区划分存在的孤立性社区问题, 提出了一种结合属性标签的拓扑势社区检测算法 (TPCDLP). 首先, 将结合标签传播思想将属性信息构造出节点间的链接权值. 其次, 把链接权值加入到拓扑势当中构造拓扑势场. 然后, 利用核心节点进行子群社区的划分. 最后, 利用子群社区间核心节点的距离进行社区划分.

1 相关工作

李德毅等 2008 年提出了社区检测中的拓扑势理论, 构造了一种在网络拓扑空间中构造的虚拟势场[8]. 拓扑势借鉴了数学中的拓扑学和物理中的场论思想, 将网络 G 看作一个包含 n 个节点的及其相互作用的抽象系统. 每一个结节周围存在一个作用场, 位于场中的任何节点都会收到其周围节点的影响. 但是节点的影响力随着网络距离的增加而快速衰减.

定义 1. 拓扑势场. 一个网络$G=(V,E)$, 网络所有节点$v_i, 1\le i\le n$都存在一个拓扑势$\varphi(v_i)$, 所有节点的拓扑势相互作用从而构成拓扑势场.

定义 2. 拓扑势. 给定网络$G=(V,E)$, 其中$V=\{v_1,v_2,\cdots,v_n\}$为网络节点, $E=\{(v_i,v_j)|v_i,v_j\in V, i\ne j\}$为节点边集合, 每个节点的拓扑势计算公式如下:

$$\phi(v_i)=\sum_{j=1}^{n}\left[m(v_j)\times e^{-\left(\frac{d_{ij}}{\sigma}\right)^2}\right] \tag{1}$$

其中, d_{ij}表示节点v_i与节点v_j之间的网络距离或跳数. 影响因子σ是用于控制每个节点的影响范围. $m(v_j)$表示节点v_j的质量, 可以用来描述每个节点的固有属性, 但是通过相似研究, 在本文设置为$m(v_j)=1$.

根据高斯函数的数学性质可知, 如果$d_{ij}>\lfloor 3\sigma/\sqrt{2}\rfloor$, 节点$v_i$对节点$v_j$的拓扑势影响会随着距离快衰减为 0, 由此可以忽略不计. 拓扑势场是一个短程场, 其影响范围有限. 所以在本文中设置$\sigma=0.4721$. 那么, $\lfloor 3\sigma/\sqrt{2}\rfloor=\lfloor 3\times 0.4721/\sqrt{2}\rfloor=1$, 也就是说, 网络节点只对其邻居节点有影响力.

本文首先利用了信息传播的特性将节点的属性结构In和链关系 E 转换成节点间的链接权重关系 R. 随后, 利用拓扑势将具有链接关系的网络结构转化成山脉形状的拓扑势域. 其次, 在山脉形状的立体结构中找到局部最高点, 由局部最高点出发进行子群社区的划分. 最后根据子群社区的分布情况, 将子群社区进行合

并得到社区的划分结果 C.

2 一种基于标签传播的拓扑势社区检测算法

2.1 节点间链接权值计算

拓扑势算法利用的是链接关系构造拓扑势场, 未考虑结节间的属性关系. 社区的定义是将具有链接紧密程度的节点化为一个社区, 但是结节间的属性关系同样会影响到社区划分的质量和现实场景的应用.

本文利用节点间的属性关系和链接关系构造节点间拓扑势的环境影响因子r_{ij}, 从而保证节点i和节点j之间的拓扑势能够受到环境影响因子影响. 公式如下:

$$\varphi(v_i)=\sum_{j=1}^{n}\left[m(v_j)\times r_{ij}\times e^{-\left(\frac{d_{ij}}{\sigma}\right)^2}\right] \tag{2}$$

借鉴标签传播的思想, 计算标签从节点v_i传播到节点v_j的概率$P(v_i\to v_j)$, 随后令节点v_i和节点v_j的环境影响因子$r_{ij}=P(v_i\to v_j)$.

2.2 标签传播特性

定义 3. 节点影响力. 设网络$G=(V,E)$中每个节点v_i都拥有一个影响力值, 用Inf_i表示. 由于大多网络并不是连通图, 因此本文采用文献 [13] 所提出 LeaderRank 算法, 计算节点的 LR 值.

LeaderRank 算法提到社交网络不是一个强连通图, 所以引入一个节点g (Ground Node), 与其他节点相互连接, 使社交网络变成一个强连通图. LeaderRank 算法核心公式:

$$LR_i(t+1)=\sum_{j=1}^{N+1}\frac{a_{ij}}{K_j^{\text{out}}}LR_i(t) \tag{3}$$

$$LR_i=LR_i(t_c)+\frac{LR_g(t_c)}{N} \tag{4}$$

其中, a_{ij}表示节点j到节点i是否有链接, 有为 1, 无为 0; K_j^{out}表示节点j的出度个数; N表示节点总个数; $LR_i(t)$表示i节点在t时刻的得分; t_c表示$LR_i(t)$收敛的得分; 表示t_c时刻地节点的得分; LR_i表示i节点最终的得分.

图 1 是一个小社交网络拓扑结构图, 一共有 18 个节点, 每个节点代表一个人, 每个人都有一个兴趣爱好, 将兴趣爱好分为两类, 并用两种不同的图标表示人们的兴趣爱好. 节点间的连线代表人们之间的关系. 通过上述的公式, 计算得到这个简单的社交网络数据集每个节点的节点影响力, 如表 1 所示.

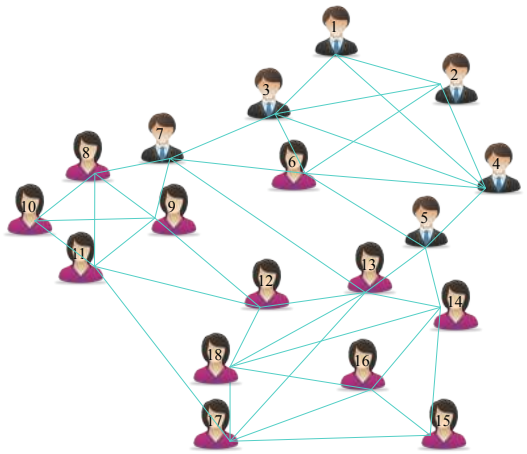

图 1 小社交网络

表 1 小社区网络的节点影响力 LR

节点ID	Inf值	节点ID	Inf值
1	0.762 913	10	0.762 707
2	0.915 510	11	1.067 720
3	1.068 080	12	0.915 160
4	1.068 070	13	1.372 680
5	0.915 294	14	1.067 610
6	1.068 030	15	0.762 559
7	1.067 890	16	1.067 580
8	0.915 294	17	1.067 580
9	1.067 780	18	1.067 580

定义 4. 传播特性k. 定义$k_{i\leftarrow j}$为标签从节点j到节点i的传播特性度量值.

$$k_{i\leftarrow j}=\frac{\log(1+LR_j)}{\log((1+LR_i)\times(1+LR_j))} \tag{5}$$

该传播特性是由节点v_i和节点v_j的节点影响力决定的. 当LR_i远大于Inf_j时, $k_{i\leftarrow j}\approx 1$, 说明$v_j$的影响力较大, 节点$v_i$容易受节点$v_j$的影响. 反之, 当$LR_j$远大于$LR_i$时, $k_{i\leftarrow j}\approx 0$, 说明$v_i$的影响力较大, 节点$v_i$不容易受节点$v_j$的影响.

以节点 1、2、3 为例, 已知$LR_1=0.762\,913$, $LR_2=0.915\,51$, $LR_3=1.068\,08$, 根据定义 4 的公式, 可得:

$$k_{1\to 2}=\frac{\log(1+0.762913)}{\log((1+0.762913)\times(1+0.91551))}\approx 0.465\,892$$

$$k_{2\to 1}=\frac{\log(1+0.91551)}{\log((1+0.762913)\times(1+1.0680))}\approx 0.534\,108$$

$$k_{1\to 3}=\frac{\log(1+0.762913)}{\log((1+0.762913)\times(1+1.0680))}\approx 0.438\,303$$

$$k_{3\to 1}=\frac{\log(1+1.0680)}{\log((1+0.762913)\times(1+1.0680))}\approx 0.561\,696$$

节点 1 的影响力LR_1小于节点 3 和节点 4 的影响力. 通过比较发现节点 1 到节点 2 的传播特性要低于节点 2 到节点 1 的传播特性, 同样的节点 1 到节点 3 的传播特性也低于节点 3 到节点 1 的传播特性.由此, 传播特性值可以反映出影响力高的节点与影响力低的节点之间受影响程度的差异.

2.3 节点间的相似度计算

社会网络不仅具有拓扑结构特征, 而且网络中节点的内在属性也容易获取, 如 C-DBLP 中的学者记录都拥有研究方向、工作单位等信息, 因此节点的属性特征S (节点的相似度) 包含两部分: 结构属性St和节点内在属性In.

$$S_{i,j} = St_{i,j} + In_{i,j} \tag{6}$$

结构属性:

$$St_{i,j} = \frac{|N(i) \cap N(j)|}{\sqrt{|N(i)| \times |N(j)|}} \tag{7}$$

节点内在属性:

$$In_{i,j} = \frac{1}{z}\sum_{k=1}^{z}\zeta(in_{ik}, in_{jk}) \tag{8}$$

$$\zeta(in_{ik}, in_{jk}) = \begin{cases} 1 & in_{ik} = in_{jk} \\ 0 & in_{ik} \neq in_{jk} \end{cases} \tag{9}$$

$N(i)$表示节点i的所有邻居与节点i的集合. $in_i = \{in_1, in_2, \cdots, in_z\}$为节点$i$的内在属性集合, in_{iz}是节点v_i的第 z 个属性值; z 是内在属性总个数.

图 1 所示的社交网络数据集中, 节点 1 和节点 2 都有一个相同的邻居节点 3 和节点 4, 所以结构属性$St_{1,2} = 2/\sqrt{3\times 4} = 2.577\,35$. 节点 1 和节点 2 节点都有相同的兴趣爱好, 所以内在属性$Ln_{1,2} = (1/2)\times(1+1) = 1$. 由此节点 1 和节点 2 间的属性特征$S_{1,2} = 0.577\,35 + 1 = 1.577\,35$. 同理, $S_{1,3} = 1.516\,40$, $S_{1,4} = 1.516\,40$.

2.4 节点间的传播概率计算

定义 5. 标签传播概率 (节点间的关联强度, 也就是边的权值). 节点j的标签以概率$P(i \leftarrow j)$传播到节点i, $P(i \leftarrow j)$取决于节点i和j的相似性度量$S_{i,j}$、传播特性度量$k_{i\leftarrow j}$和邻接矩阵$\delta(i,j)$.

$$P(i \leftarrow j) = S_{i,j} \times k_{i\leftarrow j} \times \delta(i,j) \tag{10}$$

节点j到节点i的标签传播概率体现了标签从节点j传播到节点i的能力, 也可以认为是节点j到节点i的有向边的权值. 由此可得, 节点j到节点i的有向边的权值:

$$R_{ij} = P(j \leftarrow i) \tag{11}$$

由上述公式可以计算$r_{12} = S_{1,2}\times k_{1\to 2}\times\delta(1,2) = 1.577\,35\times 0.465\,892\times 1 = 0.734\,87$, $r_{13} = 0.664\,62$, $r_{14} = 0.664\,63$. 由于节点的拓扑势公式$\varphi(v_i) = \sum_{j=1}^{n}\left[m(v_j)\times r_{ij}\times e^{-(d_{ij}/\sigma)^2}\right] = \left[\sum_{j=1}^{n} r_{ij}\right]\times e^{-(d_{ij}/\sigma)^2}$, 可以先计算节点$v_i$的$\sum_{j}^{n} r_{ij}$.如图 2 所示,节点 1 的$\sum_{j-1}^{n} r_{ij} = r_{12} + r_{13} + r_{14} = 0.734\,87 + 0.664\,62 + 0.664\,63 = 2.064\,12$. 节点 1 到节点 2 的标签传播概率决定了节点 1 将信息传递到节点 2 的能力强度, 由此也决定了节点 1 到节点 2 节点的属性信息和链接信息影响后拓扑势变化. 表 2 是将每个节点到邻居节点的环境影响因子进行加和的结果.

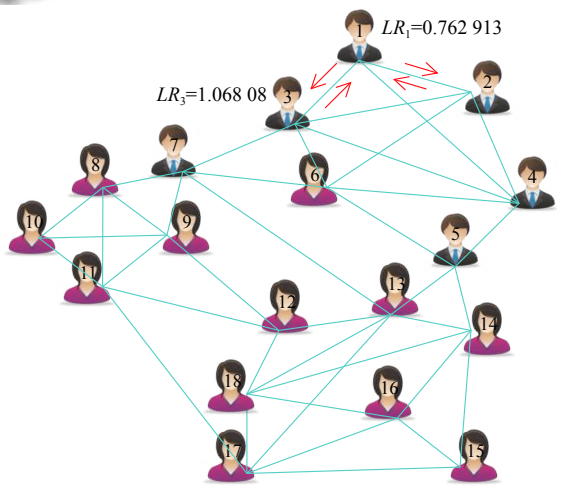

图 2 小社交网络的节点 1 的环境影响因子

表 2 小社交网络的节点环境影响因子求和

节点ID	$\sum_{j}^{n} r_{ij}$	节点ID	$\sum_{j}^{n} r_{ij}$
1	2.064 12	10	2.064 13
2	2.631 45	11	3.561 52
3	3.433 74	12	2.243 58
4	3.479 60	13	4.061 51
5	0.869 98	14	3.013 14
6	1.054 08	15	1.767 51
7	0.917 99	16	3.940 08
8	2.420 29	17	3.317 87
9	3.279 59	18	3.611 37

表 3 是通过改进后的拓扑势公式计算出图 1 的社交网络数据集的每个节点的拓扑势值. 并且将节点中拓普势局部最高的节点用五角星标记在图 3 中.

表 3 小社交网络的节点拓扑势值

节点ID	φ值	节点ID	φ值
1	0.023 236	10	0.023 236
2	0.029 622	11	0.040 093
3	0.038 654	12	0.025 256
4	0.039 170	13	0.045 721
5	0.009 783	14	0.033 919
6	0.011 866	15	0.019 897
7	0.010 334	16	0.044 354
8	0.027 246	17	0.037 349
9	0.036 918	18	0.040 654

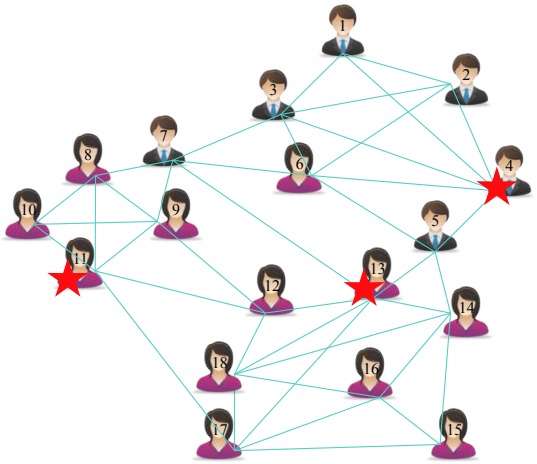

图 3 简单社交网络的拓扑势局部最高值的节点

2.5 子群社区划分

通过节点拓扑势的计算, 将网络的链接结构转变成山脉形状的拓扑势场. 社区的划分就如同山的划分, 山峰、山谷和斜坡, 对应社区的核心节点、重叠节点以及内部节点.

定义 6. 核心节点. 假设在一个社交网络$G=(V,E)$中, 其拓扑势域$G'=(V,E,\phi)$, N_i是节点v_i的邻居节点. $\forall v_j \in N_i$, 如果$\phi(v_i)>\phi(v_j)$, 则节点v_i是拓扑势域的局部最高点.

通过上述的定义, 可以看出核心节点是局部最高点, 也就是山峰节点. 如果根据当前的核心节点进行社区划分, 将会影响社区划分的质量和数量. 由此, 当前通过核心节点划分的社区被称为子群社区, 后续需要进一步处理. 图 3 中五角星标识的节点为拓扑势局部最高点, 也就是当前子群社区的核心节点.

定义 7. 重叠节点. 假设在一个社交网络$G=(V,E)$中, 其拓扑势域$G'=(V,E,\phi)$, N_i是节点v_i的邻居节点. $\forall v_j \in N_i$, 如果$\phi(v_i)<\phi(v_j)$并且节点v_i处在两个不同核心节点的社区的山谷的位置, 则节点v_i是拓扑势域的重叠节点, 也就是山谷节点.

当山谷节点就直接归属于它邻居节点所在的社区. 由此, 山谷节点i处在两个不同核心节点的社区之间, 才能被称为重叠节点.

定义 8. 内部节点. 假设在一个社交网络$G=(V,E)$中, 其拓扑势域$G'=(V,E,\phi)$, N_i是节点v_i的邻居节点. 内部节点满足下面任意一种情况成立: (1) $\exists v_j \in N_i$, 如果$\phi(v_i)<\phi(v_j)$并且$\exists v_j \in N_i$, 如果$\phi(v_i)>\phi(v_j)$, 则节点v_i处于斜波位置, 也就是拓扑势域的内部节点. (2) 如果$\phi(v_i)<\phi(v_j)$并且节点v_i处在两个同核心节点的社区的山谷的位置, 这该节点是社区的内部节点.

定义 9. 边缘节点. 假设在一个社交网络$G=(V,E)$中, 其拓扑势域$G'=(V,E,\phi)$, N_i是节点v_i的邻居节点, C_{overlap}是重叠节点的集合, $C_{\text{no-overlap}}$是不重叠节点的集合. (1) 如果$v_i \in C_{\text{overlap}}$, 则节点$v_i$是边缘节点; (2)$\exists v_j \in N_j$, 如果$v_i \in C_{\text{no-overlap}}$, 而$N_j \notin C_{\text{no-overlap}}$, 并且$N_j \notin C_{\text{overlap}}$, 则节点$v_i$是边缘节点.

边缘节点可以是社区的内部节点也可以是社区的重叠节点. 每个节点v_i都记录它到它归属社区的核心节点的最短距离NCD_i.

2.6 子群合并

在子群社区划分中, 拓扑势值为局部最大值的节点视为山峰节点, 一个山峰节点对应一个社区. 但子群社区划分中存在特殊两种情况. 1) 当社交网络数据集节点链接稀疏、节点度数相似时, 很容易导致划分社区数量过多, 社区包含节点过少等问题, 从而影响到社区的划分质量和现实应用. 2) 划分出的社区为孤立子群社区. 这种孤立的子群社区不能通过核心节点间的距离关系进行合并. 下面对两种情况给出相应解决方案.

2.6.1 子群社区划分

由于社交网络数据集的节点数多, 如果利用深度遍历的方法计算核心节点间的距离, 计算的复杂度很高, 时间耗费长, 所以为了快速得到上峰节点间的距离, 在子群社区划分的同时, 计算子群社区中每个节点到达其社区的上峰节点的距离, 最后分析了 3 种情况计算子群社区间的距离.

由于社交网络数据集的节点数多, 在子群社区划分的同时, 计算子群社区中每个节点到达其社区的山峰节点的距离, 并分析了计算子群社区间的距离的

3 种情况.

(1) 两个子群社区不重叠但边缘节点相连接

两个子群社区没有重叠节点, 但是社区间的边缘节点互联. 该情况下, 由于每个边缘节点都存储了到达它自身归属的子群社区的最短距离 *NCD*, 可以利用边缘节点进行信息交互, 得到两个子群社区的核心节点之间的距离. 但是, 边缘节点自身归属的子群社区的核心节点的距离不一定相同, 需要选取其中最短的距离为两个子群社区不重叠但边缘节点相连接的距离 *CCD*.

(2) 子群社区不重叠并且边缘节点相也不连接

子群社区的划分是根据节点的拓扑势值由高到低进行的, 但是一旦碰到当前划分的节点其拓普势值为局部最低点的时候, 也就是划分到山谷节点时, 就结束当前子群社区的划分. 为了计算不重叠且边缘节点不相连的两个子群社区的核心节点间的距离, 采用边缘节点探测方法进行计算. 即利用当前子群社区的边缘节点, 根据设置的步长向子群社区外部进行跳转. 每当跳到下一个节点, 首先判断当前节点是否归属于其他子群社区, 是, 根据跳转的步长以及初始节点和当前节点的信息计算两个社区的距离; 否, 跳转到下一个节点. 在做边缘探测的时候, 探测步长值设置为当前边缘节点到达子群社区核心节点的欧式距离的 1/2.

(3) 子群社区重叠

当子群社区之间有重叠节点, 需根据子群社区间的重叠节点到达核心节点的距离加和, 取其最短的路径长度.

对于子群社区间的距离的计算, 首先分别对上述 3 种情况进行处理和计算得到社区的最短距离, 然后将 3 种情况的结果进行比较取其最小值, 最终得到相近两两社区的最短距离.

2.6.2 子群社区合并

通过上述的 3 种情况分析和计算, 得到了相近的两个社区之间核心节点的最短路径. 根据核心节点的距离, 可以将相近的社区进行合并, 但是实际上很多数据集其节点的链接关系很稀疏, 也就是存在很多孤立的节点以及非常小的“孤立”社区, 如图 4 所示.

图 4 是 citeseer 数据集的数据节点分布, 图中显示, 左上方的节点有着紧密联系, 但下方的节点非常稀疏. 节点的稀疏易导致划分的社区数被这些稀疏分布的节点所决定, 使得社区划分范围过小失去意义. 因此在子群社区划分后, 需要将子群社区针对稀疏分布情况进行合并. 所以子群社区合并分为两种.

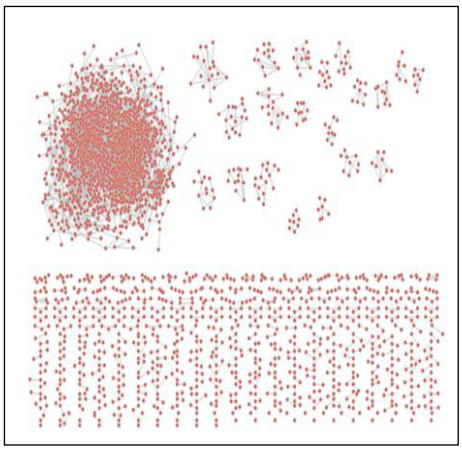

图 4 Citeseer 数据集的节点分布

(1) 相邻子群社区合并

相近的两个社区之间核心节点的最短路径存放在 *CCD*中, 计算$d = \max(CCD)$, 设置φ为合并参数取值 0-1, φd为合并距离. 当$CCD_{ij} < \varphi d$时, 将两个社区进行合并, 随机将两个子群社区中的一个核心节点设置为合并后社区的核心节点.

(2) 稀疏子群社区合并

设定规则: 核心节点的信息属性相同的稀疏子群社区合并成为一个大社区.

3 算法实验与结果分析

所有实验均在 Intel(R) Core(TM) i7- CPU 3300 和 8.00 GB RAM 的个人计算机 (PC) 上使用 Visual Studio 2015 上实现.

3.1 标准实验数据集

为验证算法有效性, 以下给出 3 个同时拥有链接和属性的社区网络数据集信息, 见表 4.

表 4 数据集信息

数据集	节点数	边数	属性
citeseer	3312	4732	6
cora	2708	5429	5
WebKB	877	1608	5

3.2 评估标准方法

为了评价 TPCDLP 算法, 本实验采用改进的模块度Q_{ov}^{E}、信息熵*Entropy*、社区重叠度*Overlap*和综合指标 *F* 作为评估指标来观测算法对社区划分的质量.

(1) 改进的模块度Q_{ov}^{E}. 由于本文是对重叠社区进行社区划分, 所以对于模块度的评估标准采用的是一种引入隶属系数的优化基础上同时发现重叠和层次社区结构的方法, 节点的隶属系数被重新定义为该节点归属社区的个数. 并且改进的模块度值越高, 说明社区内部链接更为紧密. 其具体公式如下:

$$Q_{ov}^{E}=\frac{1}{2m}\sum_{c\in C}\sum_{i,j\in c}\left[\left(A_{ij}-\frac{k_ik_j}{2m}\right)\frac{1}{O_iO_j}\right] \tag{12}$$

其中, O_i表示的是节点i所归属社团的数量, 其余和非重叠社团发现评价指标模块度Q类似.

(2) 信息熵$Entropy$. 信息熵将社区内部节点用于不相同属性的情况利用公式进行放大, 由此判断社区对于属性划分的合理性. 信息熵值越大, 说明划分出的社区内部节点拥有不同属性的情况越多, 从属性的角度分析社区划分不合理, 由此希望信息熵值小. 信息熵的公式如下:

$$Entropy=\sum_{i=1}^{z}\sum_{j=1}^{k}\frac{|c_j|}{|V|}entropy(a_i,c_j) \tag{13}$$

其中, $entropy(a_i,c_j)=-p_{ij}\log_2 p_{ij}$, p_{ij}为社区j中的节点具有属性值a_i的比例.

(3) 社区重叠度$Overlap$. 社区重叠节点的个数决定了社区重叠度$Overlap$的值. 它体现了网络耦合度, 计算公式如下:

$$Overlap=\frac{1}{m}\sum_{c\in C}|c| \tag{14}$$

其中, $|c|$表示社区c的节点个数, m表示网络节点个数.

(4) 综合指标 F. 一般情况下, 重叠度高的网络其模块度相对较低, 两者呈现负相关性.而对于实验结果而言, 模块度越大, 信息熵和重叠度越小, 社区挖掘的质量就越好. 所以综合以上情况, 为了输出更为合适的社区结果, 定义 F 值为综合评估指标:

$$F=\frac{Q_{ov}^{E}\times(Entropy+Overlap)\times 2}{Q_{ov}^{E}+Entropy+Overlap} \tag{15}$$

3.3 对比实验

3.3.1 有属性数据集实验对比

在有属性数据集实验中, 将子群社区的划分与合并进行详细的分析. 并且为了更好地展示本文提出的算法的优越性, 将本文提出的算法与 DOCET 算法、LANMF 算法、LPPB 算法[14]、Louvain 算法[15]、SCD 算法[16] 和 DEMON 算法[17] 进行实验对比. DOCET 算法、Louvain 算法、SCD 算法和 DEMON算法只考虑了社交网络数据集中节点的链接信息, 而 LANMF 算法和 LPPB 算法利用社交网络数据集中节点的链接信息和属性信息进行社区划分. 这 3 个数据集中, DOCET 算法、LANMF 算法、LPPB 算法、SCD 算法和 DEMON 算法都能进行重叠社区的划分, 而 Louvain 算法主要针对的是非重叠节点的划分.

(1) 子群社区划分

对 3 个有属性数据集进行子群社区的划分. 首先, 根据节点拓扑势值的局部最高点确定核心节点. 再利用核心节点进行子群社区划分. 最后, 将子群社区划分结果进行计算汇总, 具体如表 5 所示.

表 5 有属性数据集的子群社区划分

数据集	子群社区数	孤立子群社区数	重叠度	改进模块度	信息熵	综合指标
citeseer	498	262	1.082 125	0.684 279	0.743 478	0.995 442
cora	233	54	1.180 206	0.654 599	0.885 536	0.994 164
WebKB	35	4	1.095 781	0.819 825	1.620 860	1.259 545

如表 5 所示, 在 citeseer 数据集的子群社区数 489 个但其中有 262 个孤立的子群社区数, 也就是一半的社区是孤立子群社区. 而这些孤立子群社区节点的数量都小于 10, 由此 citeseer 数据集一半的子群社区的节点数过小. cora 数据集的子群社区数是 233 个, 其中 1/4 的子群社区是孤立子群社区. 而 WebKB 数据集子群社区数是 35 个, 它的子群社区划分数量相对于其它两个有属性数据集最小但综合指标最高. 通过表中的群社区数和综合指标的数据看, WebKB 数据集当前的子群划分效果好, 而 citeseer 数据集和 cora 数据集子群社区数多, 孤立子群社区数占子群社区数的比例大, 需要将这些数据集进行进一步的合并, 确保社区划分的综合质量.

(2) 子群社区合并

子群划分实验中, 已经将 3 个有属性的数据集进行的子群社区的划分, 接下来根据子群社区间的距离CCD和设置的合并范围φd进行社区的合并, φ的取值为 0.2, 结果如表 6 所示.

如表 5 和表 6 所示, citeseer 数据集由 498 个子群社区合并成为了 132 个社区, 是合并前子群社区数量的 1/4; cora 数据集由 233 个子群社区合并成为 45 个

社区, 是合并前子群社区数量的 1/5; WebKB 数据集由于数据量小, 合并后一共由 20 个社区, 是合并前子群社区数量的 4/7. 所以本文提出的算法在社区合并后, 3 个有属性数据集的社区数都有所下降. 而综合指标方面, citeseer 数据集由合并前 0.995442 降到 0.909849, 而 cora 数据集也由合并前 0.994 164 降到 0.876 022. citeseer 数据集和 cora 数据集合并后综合指标和合并前的综合指标差距在 0.1 左右. 然而, 造成这两个数据集在合并后综合指标下降的原因是合并子群社区后改进后的模块度下降导致. citeseer 数据集的改进模块度由合并前 0.684 279 降到 0.612 224, 而重叠度和信息熵的变化不明显. 同样的 cora 数据集的改进模块度也由合并前 0.654 599 降到 0.563 148, 而重叠度和信息熵的变化也不明显. 相反, WebKB 数据集的综合指标比合并前的综合指标高, 由合并前 1.259 545 升高到 1.309 186, 差距在 0.05 左右. 在进行子群社区的合并过程中, 综合指标在 0.1 左右浮动, 但是社区数量明显减少.

表 6 子群社区合并结果

算法	数据集	社区数	重叠度	改进模块度	信息熵	综合指标
TPCDLP	citeseer	160	1.053 4420	0.634 315	0.730 9940	0.935 9330
	cora	74	1.143 2791	**0.922 984**	0.624 6420	**1.212 7985**
	WebKB	22	1.080 9578	**0.839 338**	1.724 2900	**1.292 0815**
DOCET	citeseer	646	1.116 2430	0.613 582	0.708 6830	0.855 2560
	cora	242	1.245 1990	0.492 039	0.788 3510	0.732 3080
	WebKB	42	1.140 2500	0.721 182	1.765 4500	1.122 6110
LPPB	citeseer	10	1.211 8000	0.125 659	1.755 4200	0.241 1070
	cora	5	1.110 4100	0.180 929	1.346 5000	0.337 0380
	WebKB	5	1.052 4500	0.140 250	0.815 3190	0.260 9080
LANMF	citeseer	12	1.846 6200	0.361 819	2.423 2900	0.667 1090
	cora	10	1.509 6000	0.442 293	1.429 4800	0.768 8790
	WebKB	10	1.725 2000	0.240 839	1.796 3900	0.450 8450
Louvain	citeseer	462	1.000 0000	**0.891 144**	0.853 8530	**1.179 7870**
	cora	105	1.000 0000	0.820 324	0.587 5190	1.036 6250
	WebKB	10	1.000 0000	0.643 728	1.508 7100	0.991060
SCD	citeseer	2006	1.000 0000	0.422 828	0.231 6670	0.629 5375
	cora	1708	1.000 0000	0.313 575	0.075 3262	0.485 5571
	WebKB	156	1.000 0000	0.631 668	0.503 8900	0.889 6590
DEMON	citeseer	94	**0.295 5920**	0.229 837	**0.177 2300**	0.309 3164
	cora	125	**0.647 3410**	0.300 628	**0.307 4220**	0.457 2735
	WebKB	20	**0.508 5520**	0.178 078	**0.896 5380**	0.316 0948

表 6 中, 将文本提出的 TPCDLP 算法和其他 3 个社区检测的算法进行了比较. 通过比较可以看出, 在 citeseer 数据集中, Louvain 算法的综合指标最高, 再是本文提出的算法. 出现这种情况的原因是由于 Louvain 算法是用模块度最优的方法进行社区的划分. 所以与其他四个算法的改进模块度比较, Louvain 算法的改进模块度最高. 虽然本文用改进的模块度作为评估标准, 但是当社区为非重叠社区时, 改进的模块度计算公式其实就是模块度的公式. 所以 Louvain 算法的改进模块度相对其他算法会高, 由此综合指标也高. 然而在 cora 数据集和 WebKB 数据集中, 本文提出的算法与其他 6 个社区检测的算法比较, 改进的模块度和综合指标都是最高. 本文算法在 cora 数据集上, 改进的模块度为 0.922 984, 与其他 4 个算法的改进的模块度高出最小为 0.1 左右; 而综合指标为 1.212 7985, 与其他 6 个算法的综合指标高出最小为 0.2 左右. 本文算法在 WebKB 数据集上, 改进的模块度为 0.839 338, 同样与其他 6 个算法的改进的模块度高出最小为 0.1 左右; 而综合指标为 1.292 0815, 同样与其他 6 个算法的综合指标高出最小为 0.2 左右. 所以, 通过上述分析, TPCDLP 相对其它 6 个算法具有一定的优势.

4 真实社区应用

为了验证本文算法在现实应用中的有效性, 选择了 3 个真实社交网络数据, 如表 7 所示.

表 7 真实社交网络数据集

名称	节点数	边数	真实社区数	网络描述
Karate	34	78	2	Zachary空手道俱乐部
Dolphins	62	159	2	海豚网络
Football	115	613	12	美国大学足球

Karate 为美国空手道俱乐部跆拳道俱乐部的真实划分.

Dolphin 数据集是 D. Lusseau 等人使用长达 7 年的时间观察新西兰 Doubtful Sound 海峡 62 只海豚群体的交流情况而得到的海豚社会关系网络. 这个网络具有 62 个节点, 159 条边. 节点表示海豚, 而边表示海豚间的频繁接触, 该图为无权图.

Football 网络, 根据美国大学生足球联赛而创建的一个复杂的社会网络.该网络包含 115 个节点和 616 条边,其中网络中的结点代表足球队,两个结点之间的边表示两只球队之间进行过一场比赛.参赛的 115 支大学生代表队被分为 12 个联盟. 比赛的流程是联盟内部的球队先进行小组赛,然后再是联盟之间球队的比赛.

此处选择了标准化互信息 (NMI) 评价指标来衡量算法得到的社区划分结果与实际社区的相似性分区结果比较. NMI的计算公式如下:

$$NMI = \frac{-2\sum_{i=1}^{C_A}\sum_{i=1}^{C_B} C_{ij}\log_2(C_{ij}N/C_{i.}C_{.j})}{\sum_{i=1}^{C_A} C_{i.}\log_2(C_{ij}/N)+\sum_{i=1}^{C_B} C_{.j}\log_2(C_{.j}/N)} \tag{16}$$

其中, A 和 B 代表社区网络的两个分区, C是混淆矩阵, 混淆矩阵C中的元素C_{ij}表示社区i除以 A 和社区j除以 B 的节点数. $C_A(C_B)$ 表示 $A(B)$ 分区中的社区数, $C_i(C_j)$ 是混淆矩阵C中第i行 (j列) 元素的和, N是原始社区网络中的节点总数. 当NMI值为 1 时, 表示 A 和 B 在社区网络中的划分相同.

由于 3 个真实社区数据集不含属性信息, 此处采用 Louvain 算法、DEMON 算法和 DOCET 算法与提出的 TPCDLP 算法进行比较. 实验结果如表 8 所示: 在海豚数据集 (dolphins) 上, 本文提出的 MIFCD 算法的NMI值最高; 在空手道数据集 (karate) 上, 本文提出的 TPCDLP 算法的 NMI 值优于 DEMON 算法; 在足球数据集 (football) 上, TPCDLP 表现好于 DOCET 算法. 可以看出, TPCDLP 能够基本实现真实社区划分.

表 8 归一化互信息评价指标的实验结果

数据	Louvain	DEMON	DOCET	TPCDLP
Karate	0.687262	0.267277	0.611001	0.354738
Dolphins	0.475323	0.418992	0.406308	0.746055
Football	0.890316	0.672065	0.366826	0.519401

5 总结

本文提出了一种基于标签属性的拓扑势社区检测算法. 该算法利用标签传播方法构造节点间的链接权重, 保证分割社区中的节点具有紧密的链接, 并保持区域内部属性特征高度一致. 由于实际网络数据具有冗余关系、数据存储量大、数据分布离散等特点, 采用拓扑势最高的局部节点作为社区的核心节点进行社区划分的算法容易导致社区重叠度高、数量多, 因此, 在划分子社区之后, 利用子节点与属性特征之间的距离划分社区, 在保证社区节点之间的链接紧密性和属性相关性的同时, 能够解决细粒度独立社区问题.

参考文献

1 Fazil M, Abulaish M. A hybrid approach for detecting automated spammers in twitter. IEEE Transactions on Information Forensics and Security, 2018, 13(11): 2707–2719. [doi: 10.1109/TIFS.2018.2825958]

2 Sánchez-Oro J, Duarte A. Iterated Greedy algorithm for performing community detection in social networks. Future Generation Computer Systems, 2018, 88: 785–791. [doi: 10.1016/j.future.2018.06.010]

3 Win HN, Lynn KT. Community detection in Facebook with outlier recognition. Proceedings of the 2017 18th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing. Kanazawa, Japan. 2017.155–159.

4 Chen YL, Chuang CH, Chiu YT. Community detection based on social interactions in a social network. Journal of the Association for Information Science and Technology, 2014, 65(3): 539–550. [doi: 10.1002/asi.22986]

5 Sun XL, Lin HF. Topical community detection from mining user tagging behavior and interest. Journal of the Association for Information Science and Technology, 2013, 64(2): 321–333.

6 Zhong C, Arisona SM, Huang XF, *et al*. Detecting the dynamics of urban structure through spatial network analysis. International Journal of Geographical Information Science,

2014, 28(11): 2178–2199. [doi: 10.1080/13658816.2014.914521]

7 Girvan M, Newman MEJ. Community structure in social and biological networks. Proceedings of the National Academy of Sciences of the United States of America, 2002, 99(12): 7821–7826. [doi: 10.1073/pnas.122653799]

8 贺超波, 汤庸, 刘海, 等. 一种集成链接和属性信息的社区挖掘方法. 计算机学报, 2017, 40(3): 601–616.

9 李德仁, 王树良, 李德毅. 空间数据挖掘理论与应用. 3 版. 北京: 科学出版社, 2019.

10 淦文燕, 赫南, 李德毅, 等. 一种基于拓扑势的网络社区发现方法. 软件学报, 2009, 20(8): 2241–2254.

11 Wang ZX, Li ZC, Yuan G, Y *et al*. Tracking the evolution of overlapping communities in dynamic social networks. Knowledge-Based Systems, 2018, 157: 81–97. [doi: 10.1016/j.knosys.2018.05.026]

12 刘世超, 朱福喜, 甘琳. 基于标签传播概率的重叠社区发现算法. 计算机学报, 2016, 39(4): 717–729. [doi: 10.11897/SP.J.1016.2016.00717]

13 Li Q, Zhou T, Lü LY, *et al*. Identifying influential spreaders by weighted LeaderRank. Physica A: Statistical Mechanics and its Applications, 2014, 404: 47–55. [doi: 10.1016/j.physa.2014.02.041]

14 李东, 程鸣权, 徐杨, 等. 基于平均互信息的最优社区发现方法. 中国科学: 信息科学, 2019, 49(5): 613–629.

15 Blondel VD, Guillaume JL, Lambiotte R, *et al*. Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment, 2008, 2008(10): P10008. [doi: 10.1088/1742-5468/2008/10/P10008]

16 Prat-Pérez A, Dominguez-Sal D, Larriba-Pey JL. High quality, scalable and parallel community detection for large real graphs. Proceedings of the 23rd International Conference on World Wide Web. Seoul, Republic of Korea. 2014. 225–236.

17 Coscia M, Rossetti G, Giannotti F, *et al*. Demon: A local-first discovery method for overlapping communities. Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Beijing, China. 2012. 615–623.