

# 基于节点重要性和模块度优化的社团划分算法

张梦园, 李玲娟

(南京邮电大学 计算机学院, 江苏 南京 210023)

**摘要:**与无权网络相比,加权网络能够反映节点间关系的强弱,赋予复杂网络更加明确的物理意义,因此加权网络的社团划分具有重要的现实意义。为了提高加权网络社团划分的准确度,设计了一种基于节点重要性和模块度优化的加权网络社团划分算法 IMWCD。首先,将每个节点初始化为一个社团,并借鉴度中心性和 PageRank 的评价思想,综合节点自身信息及其邻居节点信息来计算节点重要性;然后,按节点重要性的升序遍历节点,以模块度增益最大为原则将目标节点移入相应社团中,直至各节点不需要再移动;再以各个社团为节点重新构建网络,新网络中边的权重为两个新节点对应社团的权重之和;重复以上过程,直至网络的模块度不再变化。在 LFR 人工基准网络数据集和 High-energy theory、Astrophysics 和 Condensed matter 等真实加权网络上的实验结果表明,IMWCD 算法的社团划分质量比同类型的算法有所提升,并且具有线性时间复杂度,能适用于大规模加权网络的社团划分。

**关键词:**加权网络;社团划分;节点重要性;模块度优化;BGLL 算法

中图分类号:TP311

文献标识码:A

文章编号:1673-629X(2023)04-0126-06

doi:10.3969/j.issn.1673-629X.2023.04.019

## Community Division Algorithm Based on Node Importance and Modularity Optimization

ZHANG Meng-yuan, LI Ling-juan

(School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China)

**Abstract:** Compared with unweighted networks, weighted networks can reflect strength of the relationship between nodes and give complex networks more clear physical meaning. Therefore, the community division of weighted networks has important practical significance. In order to improve the accuracy of weighted network community division, a weighted network community division algorithm IMWCD based on node importance and modularity optimization is designed. Firstly, each node is initialized as a community, and the node importance is calculated based on the information of the node itself and its neighbors by referencing the evaluation ideas of degree centrality and PageRank. Then the nodes are traversed according to the ascending order of node importance, and the target node is moved into the corresponding community based on the principle of maximum modularity gain until each node does not need to be moved again. Then, each community is taken as a node to reconstruct the network, and the weight of the edge in the new network is the sum of the weights of the corresponding communities of the two new nodes. Repeat the process until the modularity of the network does not change. The experimental results on LFR artificial benchmark network data set and real weighted networks such as High-energy theory, Astrophysics and Condensed matter show that IMWCD algorithm improves the quality of community division compared with the algorithms of the same type, and has linear time complexity. It is suitable for the community division of large-scale weighted networks.

**Key words:** weighted network; community division; node importance; modularity optimization; BGLL algorithm

## 0 引言

对复杂网络的深入研究发现,现实世界中许多系统都能抽象为网络,如人与人之间的社交网络、自然界的食物链网络、蛋白质相互作用网络等。这些网络由具有一定组织性和极大随机性的节点构成<sup>[1]</sup>。根据网

络节点之间的连边是否存在权值,复杂网络可以分为无权网络和加权网络,加权网络相比无权网络能够反映更加丰富的信息,同时赋予复杂网络更加明确的物理意义。

社团是复杂网络的一个重要特征,社团内各个节

收稿日期:2022-05-31

修回日期:2022-09-30

基金项目:国家重点研发计划专项(2020YFB2104002);江苏省重点研发计划(BE2019740)

作者简介:张梦园(1998-),男,硕士研究生,CCF 会员(K5944G),研究方向为社会网络分析;通讯作者:李玲娟,教授,CCF 会员(E200015276M),研究方向为数据挖掘、推荐系统、社会网络分析。

点间连接紧密,而社团之间只有一些稀疏的连接<sup>[2]</sup>。发现复杂网络中的社团结构能为进一步理解网络所代表的真实世界系统的结构和功能提供捷径。经典的社团划分算法有基于模块度优化的、基于标签传播的、基于局部扩展的、基于深度学习的等等<sup>[3-5]</sup>。针对加权网络的社团划分算法还相对较少,通常考虑引入边的权值来对已有的经典的无权网络划分算法进行改进。典型的基于模块度优化的加权复杂网络的社团划分算法有 WGN<sup>[6]</sup>、WFN<sup>[7-8]</sup>、CNM<sup>[9]</sup>、BGLL 等。

其中的 BGLL<sup>[10]</sup>算法分为两个阶段,第一阶段不断将节点移入(即并入)使模块度增益最大的社团,直至节点的移动不会再导致模块度增加;第二阶段将得到的社团作为新的节点对网络进行重构,在重构的网络上重复第一阶段。BGLL 算法在第二阶段对网络重构,极大地缩小了网络规模,算法的迭代会越来越快,因此具有很好的时间效率,但由于第一阶段迭代遍历节点是无序的,没有考虑节点自身在网络中的地位,因此可能会使最终划分结果的准确度不佳,仍有改进的空间。

该文考虑在无向加权网络的社团划分算法 BGLL 中引入节点重要性来调整其遍历次序,以提高算法划分结果的准确度。首先,设计了一种基于节点自身信息及其邻居节点信息的加权网络节点重要性计算方法 WNI (weighted network node importance);然后,融合 WNI 和 BGLL 算法,设计了一种基于节点重要性和模块度优化的加权网络社团划分算法 (weighted network community division algorithm based on node importance and modularity optimization),命名为 IMWCD。IMWCD 算法将 BGLL 算法每轮合并后重构的网络中的所有节点按照重要性升序排序,并作为下一轮遍历的顺序,以此为基础实现加权网络的社团划分。

## 1 相关知识

### 1.1 网络的图表示

一个真实的加权网络可以抽象表示为图  $G = (V, E, W)$  的形式,其中  $V$  表示节点集,  $E$  表示边集,  $W$  表示边的权值,节点数  $n = |V|$ ,边数  $m = |E|$ ,  $E$  中每一条边都有  $V$  中的一对点与之对应,  $w_{ij}$  表示任意有边相连的节点  $i$  和节点  $j$  之间的权值。对于无向网络而言,任意点对  $(i, j)$  和  $(j, i)$  表示同一条边,对应边的权值  $w_{ij} = w_{ji}$ ;对于有向网络,节点之间的连边具有方向性,如  $(i, j)$  唯一表示由节点  $i$  指向节点  $j$  的边。与网络中某个节点  $v$  直接相连的边的数目称为该节点的度  $d_v$ 。

### 1.2 节点重要性

有研究表明,网络中重要性较大的节点相比其他节点更能影响网络的结构和功能<sup>[11]</sup>,重要性越大的节

点越倾向于成为社团的中心。因此,节点重要性度量方法对于发现网络的社团结构具有重要的意义。评价网络中节点重要性的方法有很多,如介数中心性、度中心性、PageRank 等。基于网络全局属性的介数中心性评价方法认为:经过某个节点的最短路径数量越多,则该节点越重要<sup>[12]</sup>,这种评价方法需要计算所有节点之间的距离,时间复杂度过高。度中心性是最简单的节点重要性评价指标,直接反映与当前节点相连的邻居数量(即节点的度),节点的度越大,与之直接相连的邻居节点越多,该节点越重要<sup>[13]</sup>;这种评价方法具有计算复杂度低的优点,但只考虑了待评估节点的局部信息,没有考虑邻居节点的信息或节点在网络中所处环境等因素,可能导致计算结果误差较大。PageRank<sup>[14]</sup>是谷歌搜索引擎的核心算法,考虑的是待评估页面的邻居页面信息对待评估页面重要性的影响,认为如果一个页面被大量其他页面所指向,则此页面是重要的。

### 1.3 BGLL 算法

BGLL 算法是一种层次性贪婪算法,基于模块度优化的思想,自底向上地不断合并网络中的社团,取模块度最大时的划分状态作为最终的划分结果。BGLL 算法首先将网络中每个节点初始化为一个社团,社团编号为节点编号。之后的工作分为两个阶段。

第一阶段:对于每个节点,考察其所有邻居节点,尝试将此节点移入其邻居节点所在社团,并计算移入后的模块度增益,如果计算得出的所有增益为负,则节点仍处于原始社团,否则选择将节点移入模块度增益最大的社团内。遍历网络中所有节点,直到没有节点移动,即移动任意节点都不会再导致模块度的增加,此时第一阶段结束。其中,模块度增益定义如下:

$$\Delta Q = \left[ \frac{\sum_{in} + k_{i,in}}{2m} - \left( \frac{\sum_{tot} + k_i}{2m} \right)^2 \right] - \left[ \frac{\sum_{in}}{2m} - \left( \frac{\sum_{tot}}{2m} \right)^2 - \left( \frac{k_i}{2m} \right)^2 \right] \quad (1)$$

其中,  $\sum_{in}$  表示社团内部所有边的权值之和,  $k_{i,in}$  表示节点  $i$  与所在社团内的节点的连边的权值之和,  $k_i$  为节点  $i$  在网络中的所有连边的权值之和,  $\sum_{tot}$  表示节点  $i$  所在社团内的各个节点与网络中所有节点连边的权值之和,  $m$  为网络中所有边的权值之和。

第二阶段:将网络图进行重构,规则如下:将第一阶段划分出的社团作为新的节点,社团编号作为新的节点编号。新的节点之间连边的权值为两个原社团之间连边的权值之和,处在同一个社团内的节点使得新形成的节点有指向自己的自环边,且权值为社团内所有连边的权值之和。将第二阶段重构的网络图作为第

一阶段的输入重新进行迭代,直至网络不再改变即模块度最大时算法停止。

## 2 加权网络节点重要性计算方法 WNI 设计

如前文所述,节点重要性能反映节点在网络中的地位,重要性越大的节点越有可能成为社团的中心。合理使用节点重要性可以使社团划分算法的划分过程更具有方向性和目的性,从而获得更准确的划分结果。

该文借鉴度中心性和 PageRank 的评价思想,对加权网络中的节点重要性做出如下的量化定义:

$$I_p = I_{p'} + I_N \quad (2)$$

其中,  $I_p$  表示节点  $p$  的重要性,  $I_{p'}$  表示根据节点  $p$  自身权值信息计算得到的重要性,  $I_N$  表示根据节点  $p$  的邻居节点信息计算得到的重要性。

$I_{p'}$  的定义如下:

$$I_{p'} = \sum_{i \in \|N(p)\|} w_i' \quad (3)$$

其中,  $\|N(p)\|$  表示节点  $p$  的邻居节点数量,  $w_i'$  表示节点  $p$  的邻边中第  $i$  条边的归一化权值,定义如下:

$$w_i' = \frac{w_i - w_{\min}}{w_{\max} - w_{\min}} \quad (4)$$

其中,  $w_{\min}$  表示网络中边的最小权值,  $w_{\max}$  表示网络中边的最大权值。归一化可以防止网络规模过大时,直接计算边的权值之和可能导致的数据溢出问题。

$I_N$  的定义如下:

$$I_N = \sum_{j \in N(p)} \frac{w_{jp}}{\sum_{k \in \|N(j)\|} w_k} \quad (5)$$

其中,  $w_{jp}$  表示节点  $j$  和节点  $p$  之间的权值,  $\|N(j)\|$  表示节点  $j$  的邻居节点数目,  $w_k$  表示节点  $j$  的邻边中第  $k$  条边的权值。

以图 1 所示的简单加权网络为例,其中节点  $v_1$  拥有最多的连边且与之相连的边的权值在整个网络中占比最大,该节点应该是网络中最重要的节点。对于节点  $v_2$  和  $v_3$ ,除了考察其自身的  $I_{p'}$ ,它们的共同邻居  $v_1$  对节点  $v_3$  的重要性贡献较大,则  $v_3$  应比  $v_2$  拥有更高的重要性值。

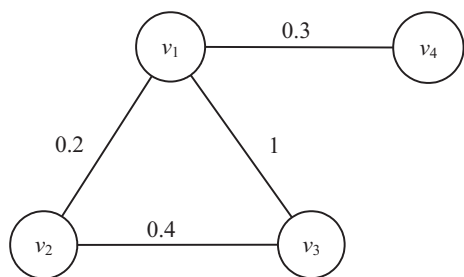


图 1 简单加权网络

用 WNI 方法计算节点的重要性。网络中边权值的最大值  $w_{\max} = 1$ , 最小值  $w_{\min} = 0.2$ 。对于节点  $v_1$ ,

$I_{v_1} = (0.2 - 0.2)/(1 - 0.2) + (1 - 0.2)/(1 - 0.2) + (0.3 - 0.2)/(1 - 0.2) = 1.125$ ;  $I_N = 0.2/(0.2 + 0.4) + 1/(1 + 0.4) + 0.3/0.3 = 2.048$ ;  $I_{v_1} = I_{v_1'} + I_N = 3.173$ 。同理,  $I_{v_2} = 0.53$ ,  $I_{v_3} = 2.58$ ,  $I_{v_4} = 0.325$ 。

可见,节点  $v_1$  的重要性  $I_{v_1}$  最大,计算结果与实际相符,说明该文设计的加权网络的节点重要性计算方法 WNI 是合理有效的。

## 3 IMWCD 算法设计与分析

如前文所述,BGLL 算法第一阶段节点的遍历次序随机,没有考虑节点自身属性信息及其所处环境信息,从而可能使最终的社团划分结果的准确度不够高。为了解决这个问题,该文设计了基于节点重要性和模块度优化的加权网络社团划分算法 IMWCD。

### 3.1 设计思路

IMWCD 采用 WNI 方法计算节点重要性,将 BGLL 算法每次迭代中第一阶段的节点遍历顺序由随机遍历调整为按节点重要性的升序遍历,以便综合利用全局和局部信息来提升算法的准确度。

### 3.2 IMWCD 算法总体流程

算法总体流程可描述如下:

输入:网络图  $G = (V, E, W)$ ;

输出:加权网络社团。

算法步骤:

Step1:初始化网络中每个节点  $i \in V$  为一个社团,社团编号为节点编号;

Step2:根据式(2)至式(5)计算每个节点的重要性,并将所有节点按照重要性升序排列成节点序列 list;

//迭代更新阶段:

Step3:考察 list 中的首个节点;

Step4:遍历当前节点的邻居节点,计算当前节点移入其邻居节点所在社团后的模块度增益,记录模块度增益的最大值和对应的社团编号;

Step5:若最大模块度增益大于零,将当前节点移入模块度增益最大的社团;否则当前节点仍留在原始社团。继续考察 list 中的下一个节点;

Step6:重复 Step4、Step5,直至当前网络中没有节点需要移动;

Step7:计算 Step6 结束后整个网络的模块度,并与上一轮的网络模块度对比,如果网络模块度未增大,则当前的划分结果为最终划分结果,算法终止;否则,执行 Step8;

//网络重构阶段:

Step8:将 Step6 中检测出的每个社团分别作为一个节点,对网络进行重构,将社团内的所有边的权值之和作为新节点指向自身的自环边的权值,新节点之间的权值为两个原社团之间连边的权值之和。将重构网络作为输入,重复 Step2 至 Step7 进行新一轮迭代。

IMWCD 算法的具体流程如图 2 所示。

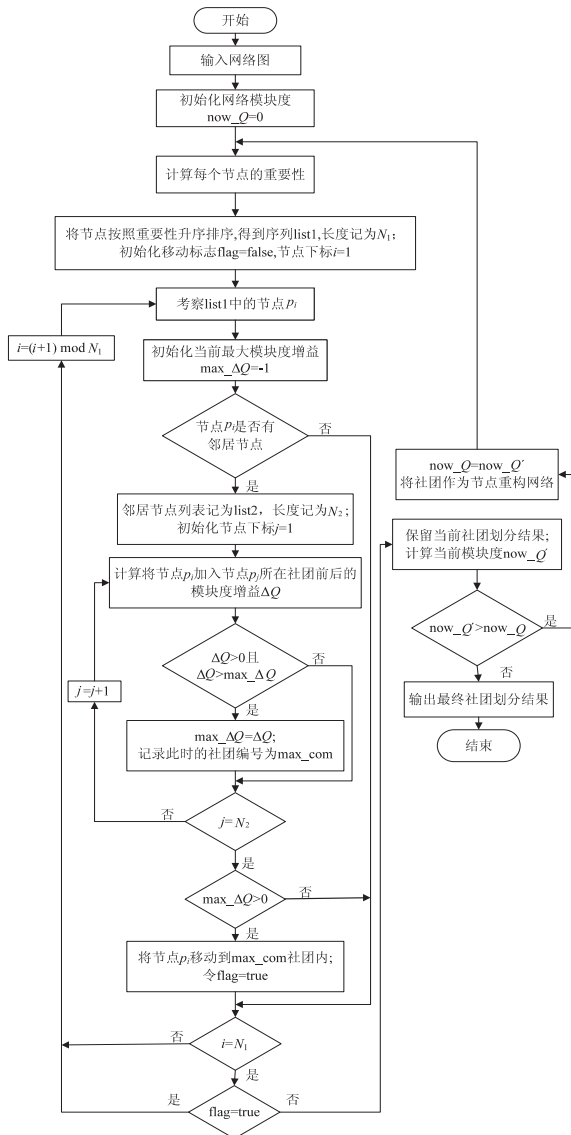


图2 IMWCD算法的具体流程

### 3.3 IMWCD 算法分析

#### (1) 实例分析。

在加权网络图中,权值代表了不同节点间联系的紧密程度,而根据社团结构的定义,社团内部节点间联系紧密,社团之间联系稀疏。以图3所示的简单加权网络为例来模拟IMWCD算法对加权网络社团结构的划分过程,检验其社团划分效果。

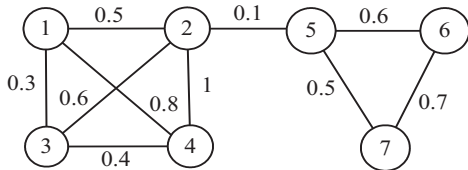


图3 加权网络

用公式(2)至公式(5)计算每个节点的重要性,得到  $I_1 = 2.266$ ,  $I_2 = 3.312$ ,  $I_3 = 1.753$ ,  $I_4 = 3.373$ ,  $I_5 = 1.942$ ,  $I_6 = 2.306$ ,  $I_7 = 2.066$ , 按照节点重要性升序排列得到节点序列  $L = \{3, 5, 7, 1, 6, 2, 4\}$ 。

将网络中的7个节点初始化为7个单独的社团,编号与节点序号对应,分别为  $C_1$  至  $C_7$ 。

迭代更新阶段:首先考察节点序列  $L$  中的节点3(称为目标节点),其邻居节点所在社团集合为  $\{1, 2, 4\}$ , 根据公式(1)分别计算将节点3移入其邻居节点所在社团  $C_i$  前后的模块度增益  $\Delta Q_i$ , 分别为:  $\Delta Q_1 = 0.0202$ ,  $\Delta Q_2 = 0.0618$ ,  $\Delta Q_4 = 0.0255$ , 则选择将节点3移入社团  $C_2$ 。同理,依次遍历剩下的节点,以模块度增益最大且为正值为原则,将节点移入相应社团,具体情况如表1所示。

表1 第一次遍历的情况

目标节点	邻居节点社团号	移入社团 $C_i$ 的增益 $\Delta Q_i$	选择移入的社团号	社团组成的变化
3	$C_1, C_2, C_4$	$\Delta Q_1 = 0.0202$ $\Delta Q_2 = 0.0618$ $\Delta Q_4 = 0.0255$	$C_2$	$C_2(2,3)$
5	$C_2, C_6, C_7$	$\Delta Q_2 = -0.0512$ $\Delta Q_6 = 0.0833$ $\Delta Q_7 = 0.0671$	$C_6$	$C_6(5,6)$
7	$C_6$	$\Delta Q_6 = 0.1686$	$C_6$	$C_6(5,6,7)$
1	$C_2, C_4$	$\Delta Q_2 = 0.0529$ $\Delta Q_4 = 0.0873$	$C_4$	$C_4(1,4)$
6	$C_6$	$\Delta Q_6 = -0.1708$	留在原始社团 $C_6$	
2	$C_4, C_6$	$\Delta Q_4 = 0.0727$ $\Delta Q_6 = -0.1782$	$C_4$	$C_2(3)$ $C_4(1,2,4)$
4	$C_2, C_4$	$\Delta Q_2 = -0.1636$ $\Delta Q_4 = -0.1491$	留在原始社团 $C_4$	

由表1可知,第一次遍历产生了三个社团,分别为  $C_2(3)$ ,  $C_4(1,2,4)$ ,  $C_6(5,6,7)$ 。因为有节点移动,所以再次从节点序列  $L$  的节点3开始新的遍历,其当前所属社团为  $C_2$ , 邻居社团只有  $C_4$ , 将其移入  $C_4$  的模块度增益  $\Delta Q_4 = 0.1074$ , 因为  $\Delta Q_4 > 0$ , 所以将其移入社团  $C_4$ ; 继续依次遍历  $L$  中其余节点, 计算得出这些节点的移动不会使模块度增加(具体计算过程不再赘述), 因此在本轮遍历过程中节点3移入  $C_4$ ,  $C_2$  消失,  $C_6$  不变。即结果有2个社团, 分别是  $C_4(1,2,3,4)$ ,  $C_6(5,6,7)$ 。

因为有节点移动,所以要继续从节点序列  $L$  的节点3开始新的遍历,再次执行迭代更新,计算得出所有节点变动后的模块度增益均小于零(具体计算过程不再赘述),即所有节点不需再移动,迭代更新阶段结束,划分出的社团结构如图4所示。此时的网络模块度  $Q = 0.4282$ 。

网络重构阶段:将迭代更新检测出的社团作为节点,对网络进行重构,重构的网络如图5所示。为了便于理解,新网络的节点号用对应的社团号表示,两个节



点分别是  $C_4$  和  $C_6$ 。

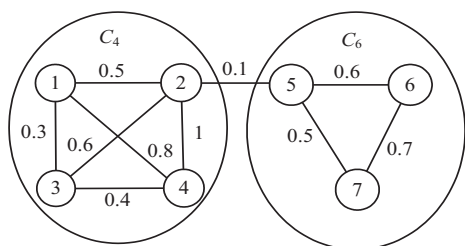


图 4 迭代更新结果

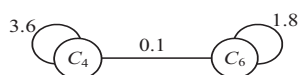


图 5 网络重构结果

对重构后的网络,用公式(2)至公式(5)计算每个节点的重要性得到  $I_{C_4} = 2.026$ ,  $I_{C_6} = 1.460$ ,升序排列的节点序列为  $L = \{C_6, C_4\}$ ,重复迭代更新阶段,此时两个节点互为邻居,  $C_6$  移动到  $C_4$  所在社团后的模块度增益是  $\Delta Q_{C_6} = -0.4283$ ,  $C_4$  移动到  $C_6$  所在社团后的模块度增益是  $\Delta Q_{C_4} = -0.4283$ ,都小于零,即都不需再移动,迭代更新结束,此时网络模块度为  $Q' = 0.4282$ ,与上一轮迭代更新结束时的网络模块度  $Q$  相同,算法终止。最终划分出的两个社团编号分别为  $C_4$  和  $C_6$ 。不难看出节点 4 和节点 6 在最初的节点重要性升序序列中确实处于高位,因此最终划分的社团以节点 4 和 6 为中心,符合实际情况,验证了 WNI 和 IMWCD 的有效性。

## (2) 时间复杂度分析。

在节点重要性计算过程中,根据节点自身权值信息计算重要性的时间复杂度为  $O(dn)$ ;  $d$  为网络的平均度值,  $n$  为网络中的节点数;基于邻居节点信息计算重要性的时间复杂度为  $O(dn)$ ,按节点重要性升序排序节点的时间复杂度为  $O(n)$ ,因此对于节点重要性计算以及节点重要性排序的算法复杂度为  $O(dn + dn + n)$ ,即  $O((2d + 1)n)$ ;传统 BGLL 算法考察每个节点的邻居节点,时间复杂度为  $O(dn)$ ,计算节点移动的模块度变化的时间复杂度为  $O(n)$ ,移动节点的复杂度为  $O(n)$ ,网络重构阶段的时间复杂度为  $O(m)$ ,  $m$  为网络中的边数。因此 BGLL 算法的时间复杂度为  $O(dn + n + n + m)$ ,即  $O((d + 2)n + m)$ 。实际大规模网络图为稀疏图,因此 BGLL 的时间复杂度近似  $O((d + 2)n)$ 。综上,IMWCD 算法的总复杂度为  $O((2d + 1)n + (d + 2)n + m)$ ,近似为  $O(3(d + 1)n)$ ,算法总的时间复杂度仍然是接近线性的。

## 4 性能测试实验及结果分析

该文分别在 LFR 人工基准网络数据集和三个真实加权网络数据集上对所设计的 IMWCD 算法的性能

进行测试,并与 BGLL 算法和 CNM 算法做对比。其中,CNM 算法首先将有  $N$  个节点的网络初始化为  $n$  个社团,然后计算社团两两合并前后的模块度增益,用堆结构来构造模块度增量矩阵,选择模块度增益最大的社团进行合并,并更新模块度增量矩阵,直至网络中所有的节点都归到一个社团。该算法针对稀疏网络的时间复杂度为  $O(m \log n)$ ,其中  $m$  为边数,  $n$  为节点数。

### (1) 实验数据集。

#### (a) LFR 人工基准网络数据集。

实验使用的 LFR 人工基准网络数据集参数见表 2。

表 2 LFR 基准网络数据集参数

网络	$N$	$k$	maxk	maxc	minc	$u$
$N_1$	1 000	30	50	50	10	$[0, 0.4]$
$N_2$	2 000	40	60	80	20	$[0, 0.4]$

表中,  $N$  表示网络中的节点数量;  $k$  表示节点的平均度;maxk 表示节点的最大度值;maxc 表示最大社团规模;minc 表示最小社团规模;  $u$  表示混合系数,  $u$  越大,社团结构越不明显。为了能够对加权社团划分算法做验证,在数据集生成过程中,对于同一个社团内的连边随机生成  $[0.5, 1]$  的权值,不同社团之间的边随机生成  $(0, 0.5)$  的权值。

#### (b) 真实网络数据集。

实验使用的三个较大规模真实网络数据集分别是 1995 至 1999 年间高能物理研究领域 (High-energy theory) 和天体物理研究领域 (Astrophysics) 的科学家合作网络,以及 1995 至 2005 年间凝聚态 (Condensed matter) 研究领域的科学家合作网络。以下分别简称 H-th、A-ph、C-mat。边的权值表示科学家们的合作次数。数据集的统计特征如表 3 所示。

表 3 数据集统计情况

真实网络	节点数 $n$	边数 $m$	平均度 $d$
H-th	8 361	15 751	3.768
A-ph	16 706	121 251	14.516
C-mat	40 421	175 692	8.693

### (2) 评价指标。

用标准化互信息  $NMI^{[15]}$  来衡量算法对 LFR 人工基准网络的社团划分准确度,NMI 值越大,表明算法划分精度越高;用模块度  $Q$  来衡量算法对真实加权网络的社团划分质量,  $Q$  值越大,表明社团划分的模块化越强,划分效果越好。

### (3) 结果分析。

#### (a) 基于 LFR 人工基准网络数据集的实验结果。

使用三种算法分别对网络  $N_1$  和  $N_2$  进行划分,NMI 的计算结果如表 4 所示。

表4 三种算法的 NMI 值

网络	混合系数 $u$	算法	NMI
$N_1$	0.2	BGLL	0.98
		IMWCD	0.98
		CNM	0.96
	0.4	BGLL	0.95
		IMWCD	0.96
		CNM	0.91
$N_2$	0.2	BGLL	0.93
		IMWCD	0.95
		CNM	0.90
	0.4	BGLL	0.91
		IMWCD	0.93
		CNM	0.85

从表4可以看出,相同混合系数下,随着节点数增多,各算法的 NMI 值略有下降。相同节点数的情况下,随着混合系数的增加,网络的社团结构越来越模糊,三个算法的表现也逐渐变差,但所设计的 IMWCD 算法的 NMI 值始终高于其他两个算法,社团划分的准确性优于其他对比算法。

(b) 基于真实加权网络数据集的实验。

表5给出了三种加权网络社团划分算法在真实网络数据集 H-th、A-ph、C-mat 上分别运行十次得到的平均模块度  $Q$  值。

表5 真实加权网络数据集社团划分结果

网络	算法	模块度 $Q$
H-th	BGLL	0.869 3
	IMWCD	0.871 3
	CNM	0.868 4
A-ph	BGLL	0.760 7
	IMWCD	0.762 5
	CNM	0.745 3
C-mat	BGLL	0.791 8
	IMWCD	0.794 0
	CNM	0.768 9

从表5可以看出,IMWCD 算法在三个较大规模的真实加权网络数据集上的模块度  $Q$  值都大于其他两个算法,验证了 IMWCD 算法在划分准确度方面的优势。

## 5 结束语

考虑到节点重要性对社团划分过程具有更明确的指示作用,以及 BGLL 算法迭代更新过程中的节点遍历顺序会影响算法划分结果的准确度,设计了一种基于节点重要性和模块度优化的加权网络社团划分算法。综合节点自身权值信息和其邻居节点信息计算节点的重要性,以节点重要性的升序为 BGLL 算法的节点遍历顺序。理论分析和在人工网络及真实加权网络数据集上的实验结果证明了 IMWCD 算法在保持线性

时间复杂度的同时,克服了 BGLL 算法节点顺序敏感问题,划分结果的准确度有所提升,能够对加权复杂网络进行有效的社团划分。

## 参考文献:

- [1] BERAHMAND K, BOUYER A, VASIGHI M. Community detection in complex networks by detecting and expanding core nodes through extended local similarity of nodes[J]. IEEE Transactions on Computational Social Systems, 2018, 5(4): 1021-1033.
- [2] 张 鹏. 基于重要节点中心性的社团网络划分[D]. 济南: 山东大学, 2019.
- [3] ZHANG Y, LIU Y, ZHU J, et al. NALPA: a no-de ability based label propagation algorithm for community detection[J]. IEEE Access, 2020, 8(1): 46642-46664.
- [4] AL-ANDOLI M N, TAN S C, CHEAH W P, et al. A review on community detection in large complex networks from conventional to deep learning methods: a call for the use of parallel meta-heuristic algorithm[J]. IEEE Access, 2021, 9(1): 96501-96527.
- [5] 李 辉, 陈福才, 张建朋, 等. 复杂网络中的社团发现算法综述[J]. 计算机应用研究, 2021, 38(6): 1611-1618.
- [6] NEWMAN M E J, GIRVAN M. Finding and evaluating community structure in networks[J]. Physical Review E, 2004, 69(2): 026113.
- [7] 杨春明, 王玉金. 基于模块度优化的加权复杂网络社团发现算法分析[J]. 西南科技大学学报, 2016, 31(4): 84-89.
- [8] YAZDANPARAST S, JAMALABDOLLAHI M, HAVENS T C. Linear time community detection by a novel modularity gain acceleration in label propagation[J]. IEEE Transactions on Big Data, 2021, 7(6): 961-966.
- [9] CLAUSET A, NEWMAN M E J, MOORE C. Finding community structure in very large networks[J]. Physical Review E, 2004, 70(6): 66111.
- [10] BLONDEL V D, GUILLAUME J L, LAMBIOTTE R, et al. Fast unfolding of communities in large networks[J]. Journal of Statistical Mechanics Theory & Experiment, 2008, 2008(10): P10008.
- [11] LÜ L, ZHOU T, ZHANG Q M, et al. The H-index of a network node and its relation to degree and coreness[J]. Nature Communications, 2016, 7(1): 10168.
- [12] 许 慧. 社会网络中节点重要性度量方法研究[D]. 哈尔滨: 哈尔滨工程大学, 2020.
- [13] 柳曾雄, 施化吉, 李 雷, 等. 基于度中心性局部扩展的社区划分算法[J]. 计算机与数字工程, 2021, 49(10): 2073-2077.
- [14] 王夏夏. 重叠社团发现算法及演化预测方法研究[D]. 南京: 南京邮电大学, 2021.
- [15] FORTUNATO S. Community detection in graphs[J]. Physics Reports, 2010, 486(3/4/5): 75-174.