社会网络中基于局部信息的边社区挖掘

潘 磊[1,2],金 杰[1,2],王崇骏[1,2],谢俊元[1,2]
(1.南京大学计算机软件新技术国家重点实验室,江苏南京 210093;2.南京大学计算机科学与技术系,江苏南京 210093)

摘 要: 近年来,随着社交网络的发展,许多重叠社区挖掘算法被提出来.传统的方法都是将节点作为研究对象,而最近的一些研究表明,以边为研究对象的边社区挖掘方法相对于点社区挖掘方法来说具有更加明显的优势.因此,我们提出了基于局部边社区的挖掘算法(LLCM),利用网络中的局部信息去挖掘边社区结构.给定一条初始的边,通过不断最大化一个适应度函数来获取该边所在的局部社区,而这条初始的边可以预先通过一些排序算法进行选择.算法经过在计算机生成网络和真实网络上测试,并且同其他边社区挖掘算法进行了比较,实验结果表明 LLCM 算法获取了合理的边社区的结构.

关键词: 社区挖掘; 边社区; 局部社区
中图分类号: TP391 **文献标识码:** A **文章编号:** 0372-2112 (2012)11-2255-09
电子学报 URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2012.11.018

Detecting Link Communities Based on Local Information in Social Networks

PAN Lei[1,2], JIN Jie[1,2], WANG Chong-jun[1,2], XIE Jun-yuan[1,2]
(1. *State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, Jiangsu* 210093, *China*;
2. *Department of Computer Science and Technology, Nanjing University, Nanjing, Jiangsu* 210093, *China*)

Abstract: Recent years have seen the development of online social networks. Many algorithms have been proposed that are able to assign each node to more than a single community. The traditional approaches were always focusing on the node community, while some recent studies have shown great advantage of link community approach which partitions links instead of nodes into communities. In this paper, we present a novel algorithm LLCM (local link community mining algorithm) for discovering link communities in networks. A local link community can be detected by maximizing a local link fitness function from a seed link, which was ranked previously. The proposed LLCM algorithm has been tested on both synthetic and real world networks, and it has been compared with other link community detecting algorithms. The experimental results showed LLCM achieves significant improvement on link community structure.

Key words: community detection; link community; local community

1 引言

挖掘社区结构已经成为一个具有普遍意义的问题,在计算机科学、数学、物理、生物以及社会学等领域都有着广泛的应用,例如:web 社区挖掘、社交网络分析、犯罪网络分析、蛋白质交互网络分析、新陈代谢网络分析、基因生物网络分析,还有客户关系挖掘和在线行为分析等,因而在国内外引起学者们的不断关注[1~3].从上世纪 70 年代开始,子图分割和社区挖掘问题逐渐成为图挖掘和社会网络分析领域关注的重点,如 Kernighan-Lin 的二分算法[4],基于 Laplace 图特征值的谱平分算法[5,6],基于电阻器网络的 Wu-Huberman 算法[7],基于 Random Walk 相似度的算法[8,9].而近年来随着 Facebook、Twitter 等社交网站的崛起,这一领域的关注度已大大提升,经过许多学者的不断研究,现在已经有了一些代表性的研究成果:GN 算法[10,11],Fast Newman 算法[12],Radicchi 快速分裂算法[13],Duch 的极值优化算法[14],Guimera 的基于模拟退火的 GA 算法[15],以及许多基于模块度的优化算法,这种方法将社区挖掘问题转化为一个优化问题,进而去寻找一个目标函数的最优解[16,17].

不过近年来对于重叠社区发现的问题更加受到关注,其中一类是基于团过滤理论(clique percolation

收稿日期:2011-12-06;修回日期:2012-04-12
基金项目:国家自然基金(No.60503021,No.60721002,No.60875038,No.61105069);江苏省科技支撑计划(No.RE2010180,No.BE2011171);南京大学研究生创新基金(No.2011CL07)

theory)：Palla 的 CPM 算法[18]，T S Evans 的 Clique Graph[19]，基于极大子团合并的 EAGLE 算法[20]．另一类是基于局部信息的方法，其中代表性的算法有：Blondel 的层次快速展开算法[21]，基于随机种子扩张的 LFM 算法[22]，基于极大子团扩张的 GCE 算法[23]，基于模型的局部扩张算法[24]，还有很多这类基于局部度量的方法[25,26]．其他算法还包括对 GN 的算法进行改进从而可以挖掘重叠社区的 CONGA 算法[27]，将信息论编码于社区划分优化的 InfoMap[28]算法，以及一些基于概率模型的方法[29,30]．

在现实网络中，尽管一个节点会属于多个不同的社区，但这个节点属于不同社区的边在社区内部还是很容易分辨出来．因此，对于高度重叠网络来说，普通的重叠社区挖掘算法不足以应付，利用边进行社区划分的方法从而被提出：Evans 将原始网络转换成线图，然后用基于随机游走的方式挖掘社区[31]，Ahn 为我们揭示了这种边社区在真实网络中的普遍存在性，在对 11 个真实数据集分别做了实验后，给出了边社区这种特殊结构的重要性[32]．而最近也有研究者将一些传统的基于节点的方法改进成基于边的形式，Kim 等人将 map equation[28]利用到了边社区的挖掘上[33]．边社区的最大优势就在于利用这种特殊的社区结构去挖掘高度重叠的社区结构．

本文也运用边社区思想，提出了基于局部信息的边社区挖掘算法，类似于节点型的局部社区(Local community)，但不同的是我们将边作为分析对象，考虑每条边在网络中所归属的局部边社区．通过一条由排序算法给定的初始边，根据适应度函数由初始边开始不断的扩张吸收周围新的边进入社区，从而得到一个局部边社区结构，然后不断地迭代这一过程即可获得覆盖网络中所有边的多个局部社区，进而得到全局的边社区结构．它的最大优势在于既体现了边社区的特殊结构，又将网络中的局部信息运用了进来，得到了较为合理的社区挖掘结果．算法在边社区的挖掘和重叠社区的覆盖上都有较好的表现．本文的主要创新点如下：

(1)放弃传统社区挖掘，图挖掘等使用节点作为研究对象，改为使用边作为研究对象；(2)改进了其他边社区挖掘方法对于局部社区的挖掘性不够的缺点，配合边社区定义，提出了一种基于局部信息的边社区发现算法；(3)提出了一种基于边聚类系数的排序方法，来选择初始种子边，可以更好的控制局部社区的分布；(4)提出了一种局部社区的层次化优化方法，从而提升边局部社区的结构合理性．

2 基于局部信息的边社区结构

给定一个网络 $G=(V,E)$，V 表示节点集合，E 表示边集合．社区发现的目的就是找出网络中符合一定条件的一些集合，这是一个 NP-hard 问题．本文将社区看作是网络中一些边组成的集合，我们的分析对象是网络中的边，那么将相似的边聚集在一个社区中就是边社区挖掘的目的．

2.1 社区的基本定义

至今为止还没有一个被公认的社区定义，一个比较普遍的定性的认识是：社区是一些节点所组成的子集，而它的内部节点之间的联系比社区之间的联系要紧密．而对于定量上定义，Radicchi 给出了一个答案[13]，他认为社区就是一个子图 S，如果 S 符合以下条件，它就是一个强社区：

$$d_i^{in} > d_i^{out}, \forall\, i \in S$$

其中，i 是任意节点，d 是节点的度数．在一个强社区中，每个节点的内部连接都要多于它的外部链接数．而如果 S 符合以下条件，它就是一个弱社区：

$$\sum_{i\in S} d_i^{in} > \sum_{i\in S} d_i^{out}$$

在弱社区中，V 中所有节点的内部度数之和大于外部度数之和．

2.2 重叠社区

我们知道，在真实社会网络中，一个节点是可以归属于多个社区的，一个人有同学、同事、家人不同的社交圈，所以这个人同时具有了多种社会网络，又比如一个学者同时参与了物理、计算机两个不同的研究课题并发表了属于两个不同领域的文章，于是这个学者也是同时加入了多个不同研究领域．那么，定义网络中的任意节点 i 同时归属于 m_i 个社区，两个社区 α 和 β 共享了 $s_{\alpha,\beta}^{ov}$ 个节点(定义 ov 为重叠尺度)，这样具有共享节点的社区就称作为重叠社区(overlapping community)[18]．但是在重叠社区中，往往和我们的直觉有一些冲突，一般我们认为社区内部联系比外部更紧密，连接的边也更多，而在重叠社区中，可能恰恰相反，社区间的联系也会很多(因为节点有相当一部分是重叠的，这取决于重叠的程度)[32]．如图 1 所示，方形节点同时属于了两个社区，而对于右边的社区(三角形节点)来说，这个节点的外部连接比内部还要多．所以重叠社区在定量上目前没有明确的定义，大多数都是算法型的定义，即挖掘出的社区结构都是算法的直接输出结果．

2.3 局部社区

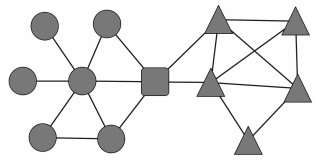

图1 一个重叠社区的例子

局部社区(这里默认是指基于节点的局部社区)本质上是一种局部结构,它包含了属于自身模块的节点以及它们周围一定范围的邻居节点.这个范围是由适应度函数 Γ 决定,在 Lancichinetti 的定义中[22],一个局部社区 S 包括节点 n_{si} 当且仅当 n_{si} 满足条件 $\Gamma(n_{si})>0$.构造一个局部社区一般是由一些初始节点开始,不断合并周围的邻居节点,最终膨胀到一个最优的规模大小,规模是由适应度函数来控制.而通过不断的发现局部社区,最终可以覆盖网络中的所有节点,从而得到一种全局的社区结构.如图 2 所示,由初始节点以及它们的邻居所构成的一个局部社区结构,方形节点是初始节点,圆形是合并后的邻居节点,三角形是待合并的候选邻居节点.另外,重叠社区的一个很重要的特点就是不同的社区之间是可以高度重叠的,重叠的程度取决于初始节点的选取,这样一来就可能需要对重叠度过高、相似度太大的两个社区进行合并.

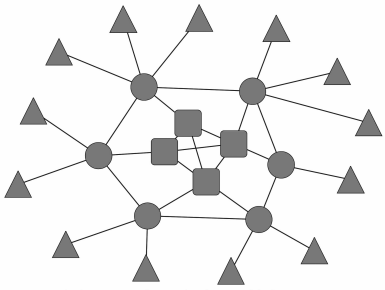

图2　一个局部社区的例子

2.4　边社区

传统的社区挖掘中是使用节点作为分析对象,发现的社区也是由一些节点以及它们所附带的边所组成的集合.而边社区主要考虑对象是边,即划分或聚集的对象都是边,通过对边的分析和挖掘,最终形成一个以边为成员的社区.例如在社交网络中,边就描绘了人与人之间的关系:社交网络上的朋友间的联系,微博上的关注等等.上文提到的重叠社区中的节点可以是“重叠的”,但是这些节点对于不同社区的连接还是很容易分辨的,节点可以属于多个社区,而边的存在一般就只有一个主要原因(同一个家庭中两个成员的连接,一起工作的人或是有相同兴趣的人等等).所以边社区试图把社区看作是一些密切相关的边的集合.如图 3 所示,圆形节点同时具有 3 个不同的社交网络:同学、同事、家庭,而边社区很好的将它们区分了开来.边社区的好处是可以方便的、自然的找出重叠社区结构同时也很容易实现,因为一个节点有同时归属于多个社区的边,我们只要将这些边划分到不同的社区中,而这个重叠节点也就自然在不同社区中同时出现.需要注意的是,在边社区中一条边只会被分配给一个社区,通过将边社区转换为节点型社区,我们可以高效地解决重叠节点和多隶属关系的发现问题,即节点根据它的多条边的不同归属从而可以属于多个社区.

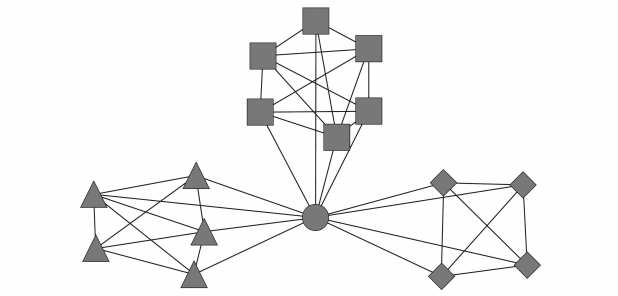

图3　一个边社区的例子

2.5　局部边社区

所谓局部边社区也就是结合了局部社区和边社区两种概念的一种社区结构,它本质上是一种边社区的结构.由一条初始选出的边开始膨胀,并通过边适应度函数的优化,得到最优的局部边社区规模,进而完成一次局部社区搜索.然后迭代的搜寻网络中的局部社区,直到最终覆盖到全网络的所有边.简单来说,局部边社区就是从网络中任意一条局部边出发所寻找出的边社区结构的集合.具体方法将在下一章节中讨论.

3　局部边社区挖掘算法

给定一个网络 G,节点数 V,边数为 E,算法 LLCM 通过:排序并选择初始边,最大化适应函数以膨胀种子边得到局部边社区,并最终获得网络中所有的边社区结构.在本节中,我们将介绍一些算法的核心步骤,包括:初始边排序与选取,边适应函数,局部膨胀,社区转换,边社区评价.

3.1　选择初始边

局部社区挖掘算法通常需要一个初始种子去挖掘社区.而初始种子的选取是很重要的.LFM[22]是随机选取种子节点.所以它的表现并不太好.GCE[23]选取了网络中的最大子团以改进 LFM 算法的不足.然后通过贪婪方法最大化评价函数.并用参数 α 控制单个局部社区的规模.但它们都是选取的节点作为初始种子.这里我们考虑用边作为初始种子.首先我们要对边进行一种排序.希望选取出密度较大的区域的代表性边.基于这种考虑我们运用聚类系数[13]对原始网络中的所有边进行排序.

定义 1(边聚类系数)　给定网络中的任意一条边 (i,j), i 与 j 分别是边的两个端点,则边聚类系数定义为

$$C_{i,j}^{(g)}=\frac{z_{i,j}^{(g)}+1}{s_{i,j}^{(g)}}=\frac{z_{i,j}^{(g)}+1}{\min(k_i-1,k_j-1)}$$

其中,$z_{i,j}^{(g)}$ 是 i,j 边所参加的多边形(边数 g)的个数,$s_{i,j}^{(g)}$ 是通过节点 i 和 j 的度数计算出的 i,j 边最多可能参与的多边形的个数.分子加 1 的目的是防止 $z_{i,j}^{(g)}$ 是 0

的时候,整个式子的值也恒为0.即使分母 $\min(k_i-1,k_j-1)=0$(此种情况下聚类系数的值不确定).

定义 2(边聚类系数 $g=3$) 在定义1的边聚类系数中,如果只考虑三角形($g=3$)的话.那么边聚类系数为

$$C_{i,j}^{(3)}=\frac{z_{i,j}^{(3)}+1}{\min(k_i-1,k_j-1)}$$

同理,$z_{i,j}^{(3)}$是 i,j 边最多可能参与的三角形的个数.当我们通过边聚类系数对所有网络中的边进行排序并取得了一个具有最大 $C_{i,j}^{(g)}$ 值的边之后,也就意味着这条边与周围邻居边的连接更加紧密,它可能是这个局部社区的核心成员之一.

3.2 边的适应度函数

在基于节点的局部社区挖掘方法里.会定义一个适应度函数(fitness function).只要选取了适当的初始种子.就可以通过贪婪算法使得适应度函数使不断增大而得到一种最优的局部社区结构[22,23].输入一个子图 S,适应度函数会返回一个实数值,这个值表示了 S 内部及 S 与外部的连接密度,增加不同的成员会引起函数值不同的变化,可能增大或减小.所以当我们有了前一步提供的初始种子后,只需要通过适应度函数来选择合适的成员,不断使这个局部社区膨胀,直到没有边可以增大适应度函数为止.Lancichinetti 等提出的适应度函数是一种代表性的局部社区的适应度函数[22],在节点型的局部社区发现上表现的是较为出色的.

定义 3(适应度) 适应度默认指的是一个社区的适应度,适应度都需要结合社区来计算.给定一个社区 S,则 S 的适应度定义为

$$f_S=\frac{k_{in}^S}{(k_{in}^S+k_{out}^S)^\alpha}$$

其中,α 是一个可以控制社区规模大小的正实数参数,社区 S 的内部和外部度数和分别是 k_{in}^S 和 k_{out}^S.值得注意的 k_{in}^S 是 S 中的内部边数的两倍整.这个函数即表达了子图内部以及外部的连接密集程度.

定义 4(节点适应度) 节点的适应度指的是节点对于一个社区的适应度贡献值,给定一个候选节点 A,它对于社区 S 的节点适应度定义为

$$f_S^A=f_{S+A}-f_{S-A}$$

这里,$S+A$ 和 $S-A$ 分别代表了加入节点 A 和没有节点 A 的子图 S,而 f_{S+A} 和 f_{S-A} 则分别表示了这两种不同情况下的社区适应度函数值,它们的差值就是节点 A 对于社区 S 的适应度贡献值.

基于以上这种思想,我们对原始的适应度函数做了修改,以适应局部边社区的需要,我们称这个新的适应度函数为边的适应度函数.

定义 5(边社区适应度) 给定一个边社区 S,则 S 的边社区适应度定义为

$$lf_S=\frac{m_{in}^S}{(m_{in}^S+m_{out}^S)^\alpha}$$

其中,α 同上定义.m_{in}^S 是子图 S 内部边数之和,m_{out}^S 是外部连接到这个子图 S 的边数之和.

定义 6(边适应度) 给定一个边社区 S 和一条候选边(i,j),则这条边关于这个社区的适应度函数为

$$lf_S^{(i,j)}=(C_{i,j}^{(g)}+2)(lf_{S+(i,j)}-lf_{S-(i,j)})$$

这里的 $S-(i,j)$ 和 $S+(i,j)$ 则分别表示原子图和加入边(i,j)后的子图的边集合.$lf_S^{(i,j)}$ 表示了(i,j)这条边对局部社区的贡献度即这条边的边适应度.在计算这个值的时候.$C_{i,j}^{(g)}$ 反应了该边所在的区域的密集程度.如果该值越大则说明此区域边越密集.我们自然应当将密集区域的边优先聚集起来.所以 $C_{i,j}^{(g)}$ 越大.它的边适应度函数值也就可能越大.当出现两条边的原始适应度函数值 $lf_{S+(i,j)}-lf_{S-(i,j)}$ 相等的时候,我们自然也就认为聚类系数大的边应更加优先被合并进入社区.另外,由于当一条边的其中一个节点的度数为0的时候,$C_{i,j}^{(g)}$ 被我们默认设置为 -1,那么为了保持整个函数式的符号不被改变,我们将前一项 $C_{i,j}^{(g)}$ 加2.

3.3 基于局部方法的边社区挖掘算法

在上一节中,已经描绘了我们算法中的几个重要的概念:边聚类系数和边适应度函数.基于以上两个部分,我们提出了基于局部方法的边社区挖掘算法,算法的整体流程如下:

(1)首先根据边聚类系数对网络中的边进行排序算法 RankLinks,并加入一个优先队列 Q 中;(2)从 Q 中取出一条值最大的种子边;(3)通过 FindLocalCommunity 算法找出种子边所在的局部边社区;(4)再重新从 Q 中取出下一条值最大的种子边,且没有被其他社区包含,重复3的过程;(5)直到所有的边都被分配到社区中为止;(6)社区结构优化,通过转换为节点型社区对社区的结构进行优化,即可得到节点型的重叠社区.

其中第一步的边排序如算法1描述.算法2则详细描述了如何由初始边形成一个局部边社区.

输入一个网络 G 以及控制规模参数 α 和聚类系数多边形参数 g,算法将通过排序边,选择种子边,膨胀社区等一系列操作最终输出一组局部边社区结构.第一步排序边操作的算法时间复杂度是 $O(m)$,其中 m 是网络中的总边数,其主要过程如算法1所描述.第二步的时间复杂度比较难计算,因为它依赖于局部社区的规模大小,这将由参数 α 决定.粗略的计算一下,对于确定的参数 α,以 e 条边构建一个局部边社区的时间复杂度是 $O(e^2)$.因此,对于整个算法来说,时间复杂度大概是 $O(m+ce^2)$,其中 c 是局部社区个数.而最坏情况

是只发现了一个局部社区且这个社区的大小就是整个网络,这时的时间复杂度是 $O(m+m^2)$.不过这种情况一般不会出现,在大多数情况下算法运行的都比较快甚至当社区规模足够小的时候可以近似到线性的时间复杂度.

算法 1　RankLinks

```
Input:一个网络 G = < V, E >
Output:网络对应的边排序队列 Q
PriorityQueue Q;
for all e∈E
   Node i = e.node1;
   Node j = e.node2;
   z3 = i.neighbors⋂j.neighbors;
   minK = min(i.degree-1, j.dgree-1 );
   c3 = z3 + 1/minK;
   z4 = neighbors(i.neighbors-{j})-{i}⋂j.neighbors;
   c4 = z4 + 1/minK;
   e.c3 = c3; e.c4 = c4;
   Q.insert(e);
end for
```

根据我们的定义,一条边的邻居边包括了这条边的两个顶点所连接的所有边.当符合条件的第一层邻居被加入社区后,第二层邻居就成为候选边,社区就如此逐层膨胀,直到社区的适应度达到最大,社区也就停止膨胀.基于边的局部社区同基于节点的局部社区有很大不同:第一,在节点型社区中增加一个节点进社区会同时附带进多条连接这个节点的边,而在边型社区中加入一条边进入只是这条边本身而不附带有任何其他边,后者也可以更精确的控制社区中各个关系的归属;第二,节点型的局部社区之间可以是互相高度重叠的,这也有可能导致"冗余社区",换句话说,同一个社区结构被这些高度相似的"冗余社区"发现了多次,这不仅浪费了计算时间和资源,而且也不符合真实网络的结构特征,还需要额外的合并算法来进行冗余消除.所以,在边型社区中,我们并不需要重叠的边社区,因为边社区中的一些节点已经是重叠的了,而且从物理上来看,边的重叠也没有什么实质性的意义.所以在我们的模型里,局部边社区彼此之间是互相独立的,它们的成员边之间是没有任何交集的.

算法 2　FindLocalCommunity

```
Input:一条初始种子边 seed
Output:其所在的局部社区
CandidatePriorityQueue Q_C;
Community C;
add seed to C;
//将第一层邻居边加入候选队列
for all e∈seed.neighbors
   if fitness(e) > 0 then
      add e to Q_C;
   end if
end for
//将 fitness 值最大的边加入社区中
A:if Q_C is not empty then
   e = Q_C.GetFront();
   add e to C;
   recalculate fitness in Q_C;
end if
else
   //重新计算候选队列中的边的 fitness 值
   for all e∈C
   if fitness(e) > 0 then
      add e to Q_C;
   end if
   end for
   //下一层邻居的 fitness 值都小于 0,算法结束
   if Q_C is empty then
      break;
   end if
   goto A;  //跳转后继续膨胀社区
end else
```

不过,由边社区转换得出的点社区不一定是全局最优的,有时候可能只是次优的.所以,为得到重叠社区结构,还需对它进行优化,优化的主要步骤如下:

(1)根据边社区与其原图的对应关系将线图划分转变成原图的覆盖(节点型社区);(2)计算当前重叠社区结构的 EQ 值[20];(3)重复如下操作直到只剩一个社区:计算每一对社区的重叠率,选择具有最大重叠率的社区对进行合并;(4)最后输出具有最大 EQ 的重叠社区.

定义 7(社区重叠度)　给定两个社区 C_1 和 C_2,则它们之间的社区重叠度(这里指的是节点型社区)为

$$O_v=\frac{|C_1\bigcap C_1|}{\min(|C_1|,|C_2|)}$$

我们发现,经过优化后的社区结构,不仅扩展模块度有所提升,而且因为社区的合并还具有了层次结构.

4　实验与结果分析

为了定量的测试我们的算法,在计算机生成网络和真实网络上分别进行了算法测试.实验环境是一台 Dell PC Intel(R) Pentiuim(R) CPU P6000 @1.87GHz 2G 内存 Microsoft Windows 7 OS.程序环境是 Java 6.0.

4.1　评价指标

对于无先验知识、无元数据、无类标的网络数据来

说,目前还没有一个通用的如 Newman 的 Q 值[11]一样的度量方式来对边社区进行评价.本文采取了目前使用较多的两种度量来对边社区结构进行评价,边连接密度(Link Density)和扩展模块度(Extended Modularity).

4.1.1 边链接密度

对于一个有条 M 边的网络,边社区 $\{P_1, P_2, \cdots, P_c\}$ 将网络划分为 C 个社区,其中社区 P_c 具有 m_C 条边,n_C 个节点,那么边社区密度定义为:

$$D_C = \frac{m_C - (n_C - 1)}{n_C(n_C - 1)/2 - (n_C - 1)}$$

$$D = \frac{1}{M} \sum_C m_C \cdot D_C$$

其中,D_C 是 m_C 被 P_C 中最大边数与最小边数的差值归一化的结果(当 $n_C = 2$ 时,设 $D_C = 0$).边社区密度 D 是 D_C 的加权平均.D 值的优势在于它并不会像 Q 值那样受到社区分辨率的限制(小于一定规模的社区无法被发现)[35],因为所有的变量都是来自于局部社区 C.边社区密度 D 度量了边社区划分的质量,当每个局部社区都是一个完全子图的时候,D 的最大值是 1,当每个局部社区都是一棵树的时候,D 的值是 0.实际上,它度量的就是边社区的"子图化"与"树化"程度.如果一个边社区的边比一棵树的边还要稀疏(如子图中有断开的部分),那么社区的连接密度将会是负值[32],当一个社区由两条断开的边组成的时候,D_C 的最小值是 $-1/3$.又因 D 是 D_C 的平均,所以 D 的下界也是 $-1/3$.

4.1.2 扩展模块度

实际上,经过优化的转换边社区可以被看作是一种重叠社区(节点的重叠),所以我们可以运用扩展模块度 EQ[20]去度量边社区的质量.众所周知,Newman 提出的模块度的思想就是试图让社区内的边足够的多,而外部连接社区的边尽量的少,也就是说当整个网络是由一些彼此独立的完全子图构成的时候,Q 值会达到最大.但是对于重叠社区来说,情况可能恰恰相反,而对重叠型社区的划分进行评价的 EQ,它对 Newman 的 Q 值进行了一些修改,我们知道在重叠节点可能对于外部的连接是较多的,而在 Newman 的定义里,这样的节点如果出现在社区内部的话,是会大大减小 Q 值的,所以 EQ 在处理重叠节点的时候,会将它们的"重叠度"考虑进去,将它对 Q 值的贡献除以节点的重叠度(—节点同时归属于的社区个数),这样一来就可以大大削弱这一类节点对于整体模块度的贡献,而使得社区内部非重叠的节点的贡献度自然而然的大大提升了.扩展的模块度(EQ)定义为:

$$EQ = \frac{1}{2m} \sum_C \sum_{i,j \in c} \frac{1}{Q_i Q_j} \left[A_{ij} - \frac{k_i k_j}{2m} \right]$$

其中,A_{ij} 是整个网络对应的邻接矩阵的任意元素(这里只考虑无向无权图的情况).若 i 与 j 有连接的话,则 $A_{ij} = 1$,否则 $A_{ij} = 0$. $m = \frac{1}{2} \sum_{ij} A_{ij}$ 表示网络中的总边数.任意节点 i 的度数为 $k_i = \sum_j A_{ij}$.节点 i 同时归属的社区数定义为 O_i.注意,当每一个节点都只归属一个社区时,EQ 就退化为了 Q,而当所有节点都归属同一个社区的时候,$EQ = 0$.毫无疑问,EQ 的值越大说明重叠社区的结构越合理、越有意义.同 Newman 的模块度一样,EQ 也受到分辨率问题的限制.

根据边社区的定义,我们发现边社区的边缘节点一定是重叠节点(只要它们与外界有连接).换句话说,它们一定同时归属于多个社区,但是在节点型的社区当中,这种情况下却不一定,边缘节点可以只属于一个社区,也可以同时属于多个社区.基于这种特殊情况,我们使用了扩展模块度与边社区密度相结合的度量方式来验证我们的社区挖掘结果.

4.2 实验数据

4.2.1 计算机生成网络

基于 Newman 模型的测试数据集[10,11]是由 128 个节点组成,其中含有 4 个大小相等的社区,每个社区中有 32 节点.网络中节点的平均度数 $\bar{K} = 16$.节点连接社区内部的边数和连接社区外部的边数和平均是 $\overline{Z_{out}} + \overline{Z_{in}} = 16$,社区内部存在一条边的概率 $P_{in} = \frac{Z_{in}}{Z_{out} + Z_{in}}$,社区间存在一条边的概率 $P_{out} = \frac{Z_{out}}{Z_{out} + Z_{in}}$,这样随机产生边时会在社区内(间)任意取两个点.但是 Newman-benchmark 的缺点是,所有的节点的度数都是相同的,社区的大小也是相同的,而且会存在一种情况,就是可能有的节点的 $Z_{out} > Z_{in}$,即使 $\overline{Z_{out}} < 8$ 的情况下,这是由它的随机性构造而造成的.另外,它只适合测试无重叠的社区发现算法.

经过对真实网络的研究,研究学者们发现网络中的节点和社区的分布是有一定统计规律的,并不是像 Newman 的模型中描述的那么简单.所以近来做重叠社区挖掘经常会用到的生成网络是 LFR-benchmark[34],它的生产网络不仅涵盖了节点和社区的统计分布规律,而且还使社区之间具有重叠和层次的效果.但是,就目前为止还没有一个可以用于边社区挖掘算法测试的计算机生成网络数据集,因为现有的计算机生成网络模型都是基于节点定义的,没有考虑到边社区的特殊结构.所以,边社区挖掘算法在现有的基于计算机生成网络上可能表现并没有基于节点的社区挖掘算法好,尽管这样,在 Newman-benchmark 和 LFR-benchmark 上我们仍然可以看到由边社区挖掘算法找出的社区具有明显的社区结构.如图 4 所示,我们可以看到我们提出的基

于局部方法的边社区挖掘算法几乎覆盖了所有的局部社区结构(圈中),它将密集区域的边都聚集在了同一社区内.

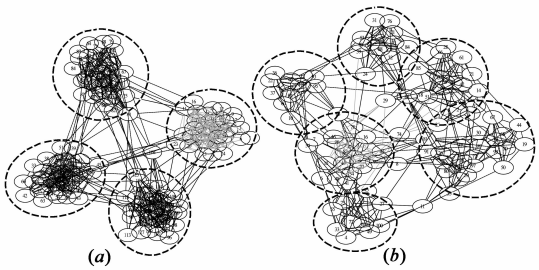

图4　计算机生成网络的边社区挖掘可视化结果

我们的算法有两个可以调节的参数:适应度函数的参数 α 以及边聚类系数的多边形参数 g,前者是用于控制局部边社区的规模大小,后者则是产生不同的初始种子节点.算法在 GN-benchmark($Z_{out}=1$)上运行的较好,虽然现在没有有效的度量方式,当从结果的可视化上来看,如图 4(a)图所示,四个不同的社区结构很明显,尽管有一些其他社区的边的干扰,但四个社区的边已经被圈形标志了出来,这些边也都聚集在了一起.而在 LFR-benchmark($n=100$, $O_n=10$, $O_m=2$, $k=10$)上的运行结果图如图 4(b)图所示,同样的我们可以看出,由 LFR 模型生成的重叠社区结构被局部边社区很好的表现出来,而且其中那些重叠节点都由它们所具有的不同归属的边展示了出来.以上实验虽然用的是基于节点的计算机生成网络模型,但就边社区的挖掘效果来看,还是很有物理意义.

由图 5 可以看出,在不同规模的 LFR 生成网络上的运行效率除了和参数 α 的选择有关,还与网络中边的规模成正比,而我们知道网络中的边数和节点数是成线性关系的($m=\bar{k}n$,其中 m 是网络中的总边数,n 是总节点数,$\bar{k}$ 是节点的平均度数),所以即使 LLCM 处理的对象是边,算法的时间复杂度并不会很高.相比之下,CPM 算法由于要对网络中的全局极大子团进行定位,使得它的实际运行时间较长.而 LC 算法针对边的相似度进行层次聚类,与 LLCM 和 CPM 相比,时间效率相对较高.

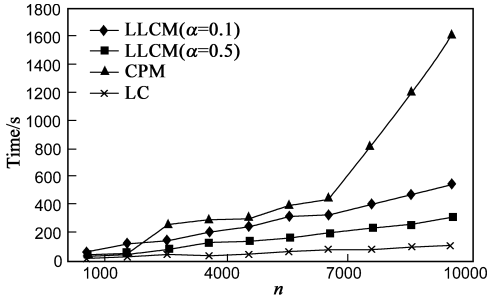

图5　运行时间与节点数的关系,横坐标为网络节点数

4.2.2　真实网络

在这一节中,将在几个真实网络数据上对我们的算法进行测试.这些网络包括:Zachary's karate club[36], Dolphins' social network[37], Books about US politics[38], and American College football[11],详细描述如表 1 所示.另外我们会使用边社区密度(D)和扩展模块度(EQ)来度量社区划分的合理性.

表 1　真实网络数据集

网络	节点	边	描述
karate	34	78	Zacharty's karate club[36]
dolphins	62	160	Dophin social network[37]
polbooks	105	441	Books about US politics[38]
football	115	613	American football[11]
netscience	1589	2742	Coauthorships in network[39]

实验结果如图 6 所示,从图中我们可以看到在真实网络下的 EQ 值比较,基于局部方法(LLCM)的边社区挖掘算法要比 CPM[18] 和 LC 算法[32] 效果略好,因为 netscience 网络大多是独立的社区组成的,所以 LLCM 和 LC 的算法结果的模块度都很高.而在 football 网络上 LLCM 的表现不如 LC 算法,那是因为这个网络是有 12 个完全不重叠的社区构成,基于层次聚类的 LC 算法更有优势.但 LLCM 挖掘出的局部社区都具有更高的扩展模块度,而边密度和 LC 相比则不足,LC 算法的 EQ 值则大多较低,并且 LC 挖掘出的社区较多,社区规模较小(CPM 算法因为无法挖掘边社区结构所以无法计算 D 值).总的来说,LLCM 挖掘的社区拓扑结构较为清晰,和 LC 算法相比更适合大社区的挖掘,其可视化结果如图 8 所示.LLCM 发现的局部社区由几种不同线形的边表示了出来,并被圈形标识了出来,而那些被不同线形的边同时连接的节点就是重叠节点.

算法调节参数 α 和 g 对结果的影响如图 7 所示,不同的 α 和 g 值影响着每个网络最终社区划分的 EQ 值.这条曲线也并不是线性,在不同网络上也不完全一样,比如在 polbooks 网络中,在 $\alpha=0.1$ 的时候取得最大的 EQ 值,但在 football 网络上 $\alpha=0.4$ 时 EQ 最大.另一个聚类系数多边形参数 g 基本上都是在 $g=4$ 时候的 EQ 值比 $g=3$ 的大,但多边形数增大也带来了计算复杂度

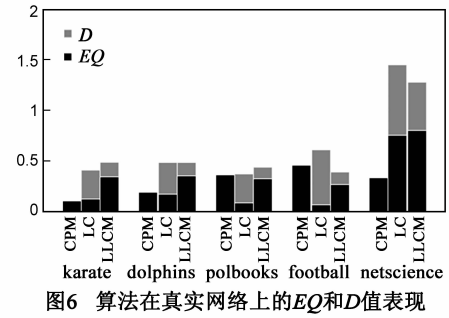

图6　算法在真实网络上的EQ和D值表现

的提升.所以在面对不同的数据集的时候,需要根据实际情况调整这两个参数以找到最适合的组合.

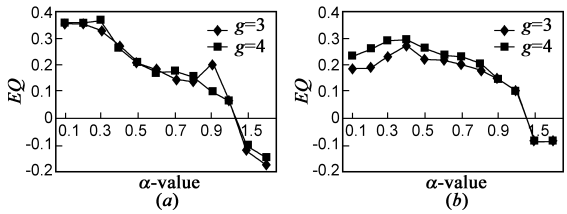

图7 参数设置对结果的影响

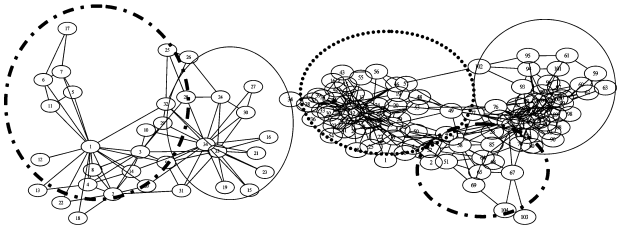

图8 真实网络的边社区可视化结果

5 结束语

本文提出了一种基于局部方法的边社区挖掘算法LLCM,该算法集中利用边而不是节点进行聚集.算法首先对网络中的边进行排序,然后选择合适的候选种子边,优化一个目标适应度函数以获得一个局部社区的最优解,通过不断迭代这一过程最终获得的是多个覆盖全网络的局部边社区结构.算法与其他边社区挖掘算法在多种计算机生成网络与真实网络上进行了比较,实验结果显示出了算法在边社区挖掘上具有一定的优势,同时算法复杂度较小,执行效率较快.但是目前,没有一个基于边社区的计算机生成网络数据集.对于没有先验知识的真实网络也没有一个被广泛认可的边社区划分的度量指标.所以我们的后续工作将在这两方面开展,定义一种基于边的计算机生成网络与社区生成模型,以及定义一种用于通用的边社区划分度量指标.

参考文献

[1] 杨博,刘大有,等.复杂网络聚类方法[J].软件学报,2009,20(1):54－66.
B Yang,D Y Liu,et al.Complex network clustering algorithms [J].Journal of Software,2009,20(1):54－66.(in Chinese)

[2] 金弟,刘大友,等.基于局部探测的快速复杂网络聚类算法[J].电子学报,2011,30(11):2540－2546.
D Jin,D Y Liu,et al.Fast complex network clustering algorithm using local detection [J].Acta Electronica Sinica,2011,30(11):2540－2546.(in Chinese).

[3] 黄健斌,孙鹤立,Dustin BORTNER,刘亚光.从链接密度遍历序列中挖掘网络社区的层次结构[J].软件学报,2011,22(5):951－961.
J B Huang,H L Sun,B Dustin,Y G Liu.Mining hierarchical community structure within networks from density-connected traveling orders [J].Journal of Software,2011,22(5):951－961.(in Chinese)

[4] B W Kernighan,S Lin.An efficient heuristic procedure for partitioning graphs [J].The Bell system technical journal,1970,49(1):291－307.

[5] M Belkin,P Niyogi.Laplacian eigenmaps and spectral techniques for embedding and clustering [A].Advances in Neural Information Processing Systems [C].Vancouver,Canada:MIT Press,2001,14:585－591.

[6] S White,P Smyth.A spectral clustering approach to finding communities in graphs [A].Kamath C,Goodman A,eds.Proceedings of the 5th SIAM International Conference on Data Mining [C].Philadelphia:SIAM,2005.76－84.

[7] F Wu,B A Huberman.Finding communities in linear time:a physics approach [J].The European Physical Journal B-Condensed Matter and Complex Systems,2004,38(2):331－338.

[8] H Zhou.Distance,Dissimilarity index,and network community structure [J].Physical Review E,2003,67(6):061901.

[9] P Pons,M Latapy.Computing communities in large networks using random walks [A].Proceedings of Computer and Information Sciences,-ISCIS 2005 [C].Berlin,Heidelberg:SpringerVerlag,2005,3733(31):284－293.

[10] M Girvan,M E J Newman.Community structure in social and biological networks [J].Proceedings of National Academy of Science of the United States of America,2002,99:7821－7826.

[11] M E J Newman,M Girvan.Finding and evaluating community structure in networks [J].Physical Review E,2004,69:026113.

[12] M E J Newman.Fast algorithm for detecting community structure in networks [J].Physical Review E,2004,69:066133.

[13] F Radicchi,C Castellano,F Cecconi,V Loreto,D Parisi.Defining and identifying communities in networks [J].Proceedings of the National Academy of Sciences of the United States of America,2004,101(9):2658－2663.

[14] J Duch,AArenas.Community detection in complex networks using extremal optimization [J].Physical Review E,2005,72(2):027104.

[15] R Guimera,L A NunesAmaral.Functional cartography of complex metabolic networks [J].Nature,2005,433(7028):895－900.

[16] D Jin,D X He,et al.Genetic algorithm with local search for community mining in complex networks [A].Proceedings of IEEE International Conference on Tools with Artificial Intelligence [C].Arras,France:IEEE,2010.105－112.

[17] C Pizzuti.A multi-objective genetic algorithm for community detection in networks [A].Proceedings of IEEE International

Conference on Tools with Artificial Intelligence [C]. Newark, New Jersey, USA: IEEE, 2009. 379 – 386.
[18] G Palla, I Derenyi, I Farkas, T Vicsek. Uncovering the overlapping community structure of complex networks in nature and society [J]. Nature, 2005, 435(7043): 814 – 818.
[19] T S Evans. Clique graphs and overlapping communities [J]. Journal of Statistical Mechanics: Theory and Experiment, 2010, 12: P12037 + .
[20] H Shen, X Cheng, et al. Detect overlapping and hierarchical community structure in networks [J]. Physica A: Statistical Mechanics and its Applications, 2009, 388(8): 1706 – 1712.
[21] V D Blondel, J L Guillaume, et al. Fast unfolding of communities in large networks [J]. Journal of Statistical Mechanics: Theory and Experiment, 2008, 10: P10008.
[22] A Lancichinetti, S Fortunato, J Kertész. Detecting the overlapping and hierarchical community structure in complex networks [J]. New Journal of Physics, 2009, 11(3): 033015.
[23] C Lee, F Reid, AMcDaid, N Hurley. Detecting highly overlapping community structure by greedy clique expansion [A]. Proceedings of SNA-KDD Workshop [C]. Washington DC, USA: IEEE, 2010. 33 – 42.
[24] A McDaid, N Hurley. Detecting highly overlapping communities with model-based overlapping seed expansion [A]. Proceedings of International Conference on Advances in Social Networks Analysis and Mining [C]. Odense, Denmark: IEEE, 2010. 112 – 119.
[25] J P Bagrow, E M Bollt. Local method for detecting communities [J]. Physical Review E, 2005, 72(4): 046108 + .
[26] F Havemann, M Heinz, et al. Identification of overlapping communities and their hierarchy by locally calculating community-changing resolution levels [J]. Journal of Statistical Mechanics: Theory and Experiment, 2011, 01: P01023 + .
[27] S Gregory. An algorithm to find overlapping community structure in networks [A]. Proceedings of PKDD 2007 [C]. Warsaw, Poland: IEEE, Lecture Notes in Computer Science, 2007, 4702(12): 91 – 102.
[28] M Rosvall, D Axelsson, C T Bergstrom. The map equation [J]. The European Physical Journal-Special Topics, 2009, 178(1): 13 – 23.
[29] B Karrer, M E J Newman. Stochastic blockmructure in networks [J]. Physical Review E, 2011, 83(1): 016107 + .
[30] B Ball, B Karrer, M E J Newman. An efficient and principled method for detecting communities in networks [J]. Physical Review E, 2011, 84: 036103.
[31] T S Evans, R Lambiotte. Line graphs, link partitions, and overlapping communities [J]. Physical Review E (Statistical, Nonlinear, and Soft Matter Physics), 2009, 80(1): 016105 + .
[32] Y YAhn, J P Bagrow, S Lehmann. Link communities reveal multi-scale complexity in networks [J]. Nature, 2010, 466: 761 – 764.
[33] Y Kim, H Jeong. The map equation for link community [J]. Physical Review E, 2011, 84: 026110.
[34] A Lancichinetti, S Fortunato, F Radicchi. Benchmark graphs for testing community detection algorithms [J]. Physical Review E, 2008, 78(4): 046110 + .
[35] S Fortunato, M Barthélemy. Resolution limit in community detection [J]. Proceedings of the National Academy of Sciences, 2007, 104(1): 36 – 41.
[36] W WZachary. An information flow model for conict and fission in small groups [J]. Anthropological Research, 1977, 33: 452 – 473.
[37] D Lusseau. The emergent properties of a dolphin social network [J]. Proceedings of the Royal Society of London, Series B: Biological Sciences, 2003, 270(2): S186 – 188.
[38] M E J Newman. Modularity and community structure in networks [J]. Proceedings of the National Academy of Sciences, 2006, 103: 8577 – 8582.
[39] M E J Newman. Finding community structure in networks using the eigenvectors of matrices. [J]. Physical Review E, 2006, 74(3): 036104.

作者简介

潘　磊　男,江苏南京人,博士生,主要研究领域为社会网络分析,数据挖掘.

金　杰　男,江苏南京人,硕士,主要研究领域为机器学习.

王崇骏(通讯作者)　男,江苏盱眙人,博士,副教授,主要研究领域为机器学习,海量数据挖掘,分布式人工智能等.
E-mail: chjwang@nju.edu.cn

谢俊元　男,1961 年生,江苏苏州人,教授,博士生导师,主要研究领域为智能系统,智能信息处理等.