Hadoop下基于边聚类的重叠社区发现算法研究

方木云,刘洪彬,谢恩文
(安徽工业大学 计算机学院,安徽 马鞍山 243032)

摘 要: 复杂网络发现算法旨在揭示网络的真实结构,对分析网络的拓扑结构、理解复杂网络的功能、寻找网络中隐藏的规律,不仅具有理论意义,而且具有广泛的应用前景。针对现有的复杂网络社区发现算法大都无法发现具有重叠性的社区结构,文中提出一种基于边的聚类算法,并且通过分布式计算的方法得到网络中节点的社区结构。实验结果表明,发现的社区结构明显优化,得到了符合真实世界的重叠社区划分。该算法能够有效发现重叠社区,运用分布式框架,在处理大规模图上实现对重叠社区的划分。

关键词: Hadoop; 边聚类; 重叠社区; 复杂网络
中图分类号: TP311 **文献标识码:** A **文章编号:** 1673-629X(2015)03-0058-05
doi: 10.3969/j.issn.1673-629X.2015.03.014

Research on Overlapping Communities Detecting Algorithm Using Hadoop Based on Edge Clustering

FANG Mu-yun, LIU Hong-bin, XIE En-wen
(College of Computer, Anhui University of Technology, Ma'anshan 243032, China)

Abstract: Traditional complex network detecting algorithm aims to reveal the true structure of network, for analyzing the topological structure of network, understanding the function of complex networks and looking for the hidden law in network, it is not only to have theoretical significance, but also wide application prospect. In the current days, complex network communities detecting algorithm mostly could not find the overlapping communities structure. In view of this problem, propose a novel clustering algorithm based on edge, which could get the communities structure of nodes in network through distributed computing. Experimental results show that the communities structure is obviously optimized, and get the overlapping communities structure which reflects the real world. This algorithm can effectively detect overlapping communities, using the distributed framework, realize the division of overlapping communities in the large graph.

Key words: Hadoop; edge clustering; overlapping community; complex network

0 引 言

现实世界中许多系统都可以用复杂网络来描述,如科研合作网、万维网、论文引用网、因特网、电力网、神经网络、新陈代谢网络以及食物链网络等。近年来,复杂网络已经成为当今科学界研究的前沿和热点。

社区结构在许多方面具有重要应用价值。例如市场营销、犯罪团伙发现,亚马逊、淘宝等推荐相似用户购买的产品,人人网、朋友网等相似特征或者兴趣爱好的用户被分为同一个社区,等等。

对复杂网络聚类算法的研究也不断增多。复杂网络聚类算法旨在揭示出复杂网络中真实存在的网络簇结构,对其的研究对分析复杂网络的拓扑结构、理解复杂网络的功能、发现复杂网络中的隐藏规律以及预测复杂网络的行为,不仅具有十分重要的理论意义,而且具有广泛的应用前景。

在以往的社区发现研究中,大多是将网络划分成相互独立的若干社区。如基于贪婪思想的Kernighan-Lin算法[1]、基于图的拉普拉斯矩阵特征向量的谱二分法[2]、基于边介数的图分割思想的GN算法[3]、基于凝聚式层次聚类的Newman快速算法[4]等。以上算法都是将网络划分为若干不相关的独立社区,但真实世界中的社交网络[5]往往是相互重叠的若干社区的叠

收稿日期: 2014-04-17 **修回日期:** 2014-07-20 **网络出版时间:** 2015-01-20
基金项目: 国家自然科学基金资助项目(61003311); 安徽省教育重大项目(ZD2008005-1)
作者简介: 方木云(1968-),男,博士,教授,研究方向为软件工程等。
网络出版地址: http://www.cnki.net/kcms/detail/61.1450.TP.20150120.2200.022.html

加,同一个节点同时属于多个社团[6]。因此以往的社区发现算法不能真实反映现实的社区划分情况。

文中提出一种基于边图的聚类算法,并且通过分布式计算方法得到网络中节点的重叠性的社区结构。

1　边图理论及相似度的构造

现实世界的网络中的节点往往属于多个社区,但两个节点之间的边代表两个节点共同拥有的属性,一个网络内不同的属性需要划分在相互独立的社区内,因此节点的边可以划分成相互独立的社区,将传统的图转化为边图,对边图进行社区划分,就将节点的重叠社区划分问题转化为了对边的非重叠社区划分问题。对划分后的边社区,根据对应关系可以求得对应节点的划分,即可得到节点的重叠社区划分。

1.1　边图理论介绍

在以往的社团结构研究中,网络图 G 通常采用节点的邻接矩阵来表示,但是邻接矩阵往往只能表示图 G 中节点和节点的相互关系,不能用来表示连边的信息。因此文中选用关联矩阵表示 G。

文中仅针对无向无权图进行研究。

定义 1: 假设给定图 $G(V,E)$,$V(G)$ 是图 G 中所有节点的集合,$E(G)$ 是图 G 中所有节点的集合。LG 是图 G 相应的边图[7],满足

条件 1: $V(\mathrm{LG}) = E(G)$;

条件 2: LG 中两顶点相邻当且仅当它们是 G 中的两条相邻的边。

则称 LG 是 G 的边图。

图 1 是由图 G 转化得到对应的边图 LG 的过程。

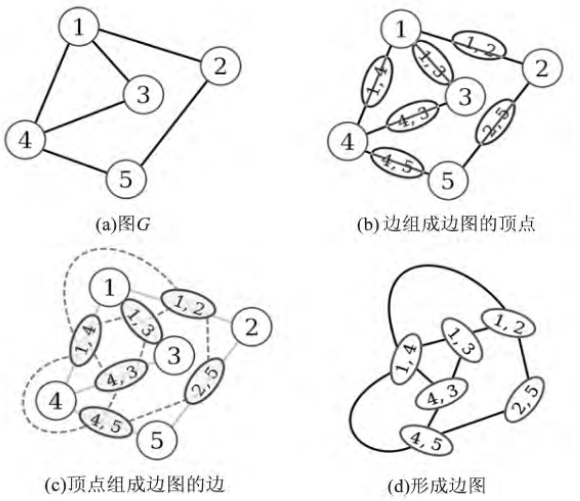

图 1　图转化为边图

关于边图的理论,经过前人的广泛研究,证明了边图具备很多优良特性。Whitney 的唯一性定理[8]更是证明了图 G 的结构完全可以从边图 $L(G)$ 中推导出来。因此将网络图 G 的关联矩阵转换成对应边图的邻接矩阵不会失去原网络图的信息。

1.2　网络图与边图的相互转化

通常网络图由节点的邻接矩阵表示,边图通常由边与边的连接矩阵表示。由网络图的连接矩阵转化为对应边图的邻接矩阵,必须先将网络图的连接矩阵转化为对应的关联矩阵,然后将关联矩阵转化为对应边图的邻接矩阵。

定义 2: 设网络图 G 的邻接矩阵 $\boldsymbol{A}$ 为 $N \times N$ 阶矩阵,则它对应的关联矩阵 $\boldsymbol{B}$ 为 $N \times L$ 阶矩阵,其中 N 为 G 中的节点数目,L 为 G 中的边的数目。若节点 i 与边 α 相互关联,则 $B_{i\alpha} = 1$;否则 $B_{i\alpha} = 0$。通过关联矩阵可以推导出节点的度 K_i 和边关联的节点数目 K_α。有

$$K_i = \sum_{\alpha} B_{i\alpha},\ K_\alpha = \sum_{i} B_{i\alpha}$$

关联矩阵和邻接矩阵的相互转化关系在文献[9]中有详细的说明。转化公式为

$$A_{ij} = \sum_{\alpha} B_{i\alpha}B_{j\alpha} - K_i\delta_{ij} \tag{1}$$

当 $i = j$ 时,$\delta_{ij} = 1$,否则 $\delta_{ij} = 0$。

由公式(1)可以根据网络图 G 的邻接矩阵 $\boldsymbol{A}$,求得其对应的关联矩阵 $\boldsymbol{B}$。

根据定义 1 和定义 2,设网络图的边图 LG 采用邻接矩阵 $\boldsymbol{C}$ 表示,则 $\boldsymbol{C}$ 是一个 $L \times L$ 阶的对称矩阵。根据公式(1)提出的邻接矩阵和关联矩阵的转换公式,可以推出由关联矩阵转换为边图的邻接矩阵的公式。

$$\boldsymbol{C}_{\alpha\beta} = \sum_{i} B_{i\alpha}B_{i\beta}(1 - \delta_{\alpha\beta}) \tag{2}$$

当 $\alpha = \beta$ 时,$\delta_{\alpha\beta} = 1$,否则 $\delta_{\alpha\beta} = 0$。

由公式(2)可以根据网络图 G 的关联矩阵 $\boldsymbol{B}$,求得其对应的边图的邻接矩阵 $\boldsymbol{C}$。

2　基于边聚类的发现算法

经过以上部分的研究,根据网络图的邻接矩阵,求得其对应的关联矩阵,然后又根据关联矩阵求得其对应的边图的邻接矩阵。网络图的节点可能有多种特性属于多个社区,但两个节点之间的边往往只代表一种特性,只能属于一个社区。因此将对节点的重叠社区划分问题转化为了对边的非重叠社区划分问题。

2.1　边结构相似度函数

在对边进行聚类的过程中,要有衡量的指标,文中选取两个边之间的边结构相似度为衡量的指标。

定义 3(边邻居): 边的邻居定义为与该边之间有节点的边。

定义 4(边结构): 边 L 的边结构定义为由该边的边邻居加上边 L 组成,表示为

$$\Gamma(L) = \{I \in E \mid (L,I) \in V\} \cup \{L\} \tag{3}$$

定义 5(边结构相似度): 两个边的结构相似度定

义为它们边结构共享的边个数与它们各自边结构数乘积的开方的商，表示为

$$\sigma(L,I) = \frac{|\Gamma(L) \cap \Gamma(I)|}{\sqrt{|\Gamma(L)| \, |\Gamma(I)|}} \quad (4)$$

根据公式(3)和(4)，可以求出任意两个边之间的边结构相似度。

图2是边图边结构相似度求解的示例。1～5分别代表5条边，两条边之间有连线代表两边之间有节点。

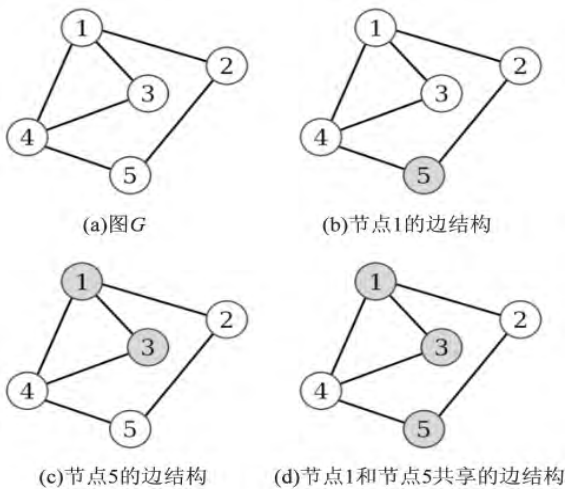

图2 边结构相似度求解过程

图(a)代表所要求边结构相似度的两个边是1和5，即 $\sigma(1,5)$；

图(b)代表边1的边结构 $\Gamma(1) = \{1,2,3,4\}$；

图(c)代表边5的边结构 $\Gamma(5) = \{2,4,5\}$；

图(d)代表边1和边5共享的边结构 $\Gamma(1) \cap \Gamma(5) = \{2,4\}$。

因此，边1和边5的边结构相似度 $\sigma(1,5) = \frac{|\Gamma(1) \cap \Gamma(5)|}{\sqrt{|\Gamma(1)| \, |\Gamma(5)|}} = \frac{2}{\sqrt{4*3}} = \frac{1}{\sqrt{3}}$。

伪代码如下：

```
getSimilarNum(F,num1,num2) {
//F是边图的邻接矩阵，num1、num2是所要求的两个边结构相似度的边号
//得到两个边的边结构
EdgeStructure1 = getEdgeStructure(F,num1);
EdgeStructure2 = getEdgeStructure(F,num2);
//得到边num1,num2的边结构的交集
EdgeStructureSum = getEdgeStructureSum(EdgeStructure1,EdgeStructure2);
//边结构相似度=边结构的交集的数量/sqrt(num1边结构的数量* num2边结构的数量)
SimilarNum = EdgeStructureSum.size()/Math.sqrt(EdgeStructure1.size() * EdgeStructure2.size());
return SimilarNum;
}
```

2.2 对边进行聚类的算法描述

(1)对边图所有边进行遍历，计算所有边及其他边的相似度总和，并对其进行排序；

(2)然后取出最大的前 K 个边作为起始聚类中心；

(3)对于所剩下的其他边，根据它们与这些聚类中心的相似度，分别分配给最相似的聚类；

(4)再重新选择每个聚类的聚类中心；这样依次进行迭代，直至两次聚类的聚类中心没有变化，才聚类结束；

(5)然后将剩下的其他边，根据它们与这些最终聚类中心的相似度，分别分配给最相似的聚类，最终得到边聚类结果。

伪代码如下：

```
//首次聚类，获得相似度总和最大的k个边，作为初始聚类中心
k_center = firstCluster(F,k);
//进行迭代，直至flag为false，或迭代次数超过上限
while(flag&&iterNum < iterTime){
//根据旧的聚类中心，重新获得新的聚类中心
k_center_new = doCluster(F,k_center_old);
//检查两次聚类的结果是否相同，相同就将flag = false
flag = check(k_center_new,k_center_old);
iterNum + +;
}
//末次聚类，根据最终聚类中心将其他边分配到相应的聚类
clusterrResult = lastCluster(F,k_center_new);
```

2.3 边聚类采用 Hadoop 下的分布式计算

Hadoop 框架是 MapReduce 模型最负盛名的开源架构实现[10]，其创始人是在互联网界鼎鼎大名的 Doug Cutting，他是 Apache Lucene 项目的创始人。2006 年他加盟雅虎，自此开始便有专业的团队开始将 Hadoop 框架的原型开发成为产品级的系统框架，2008 年，雅虎公司正式宣布它的索引正是建立在有 1 000 多个节点的 Hadoop 集群上。当年，Hadoop 框架作为 Apache 基金会的顶级项目对外开源，任何其他公司都可免费采用。用户可以在不了解分布式底层细节的情况下，开发分布式程序，充分利用集群的高速运算和存储。

MapReduce 框架的处理流程[11-13]如下：

(1)将源数据分割成 N 个可以并行计算的部分，分别放于集群中不同的节点，进行计算。

(2)将在源数据中获取的数据转化成 < key value > 对，然后输入进 map 函数。

(3) map 函数进行相关处理输出中间运算结果。

(4) Combine 函数将本地节点上 map 函数生成的中间结果进行处理、合并。

(5) 所有 map 过程进行结束后,具有相同 key 值的 value 组成一个列表。

(6) reduce 函数将根据 < key,value > 对产生相应的结果集。

图 3 是 Hadoop 的 MapReduce 框架的编程模型。

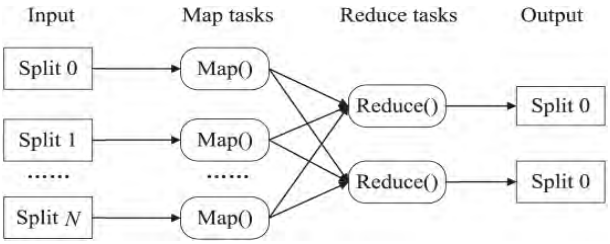

图 3　MapReduce 编程模型

2.4　边聚类的结果转化为图节点的聚类结果

边聚类最终得到的聚类结果是边的社区划分情况,需要将边聚类的结果转化为网络图节点的聚类情况才能得到文中所要求的重叠社区发现的结果。

边图中边和网络图的节点具有 1 对 2 的关系,将各个边聚类结果中的边转化为所对应的节点号,每个聚类中去除重复的节点号,即可得到所求的节点的重叠社区划分[14]。

伪代码如下:

```
for( i = 0; i < clusterrResult. size( ) ; i + + ) {
//遍历边聚类 i 结果
for( j = 0; j < clusterrResult. get( i) . size( ) ; j + + ) {
//遍历边对应的节点
for( item: edgePointRelation. get( clusterrResult. get( i) . get
( j) ) ) {
//若临时聚类列表不包含节点号,将其加入列表
if( ! temp_result. contains( item) ) temp_result. add( item) ;
}
} //遍历完聚类 i,将聚类结果加入最终聚类结果列表
result. add( temp_result) ;
}
```

3　实验与分析

文中实验采用社区发现算法研究经常使用的 Zachary 空手道俱乐部数据集[13] 进行测试。对比传统的社区发现算法,如 Kernighan – Lin 算法、谱二分法、GN 算法、Newman 快速算法等,该实验所提出的算法能够有效地进行重叠社区的发现。

在以往的社区发现算法的研究中,往往使用 Zachary 空手道俱乐部数据集。这个数据集描述的是美国一所著名大学的空手道俱乐部 34 名成员之间的相互关系。该数据集包括 34 个节点(代表 34 名俱乐部成员) 和 78 条边(代表俱乐部成员之间的关系) 。

采用传统的社区发现算法,只能将社区划分为若干个相互独立的社区。如图 4 所示,采用 GN 算法将网络划分为两个相互独立的社区。由实验结果可知,传统的社区划分方法,只能硬性地将网络划分为若干个相互独立的社区。

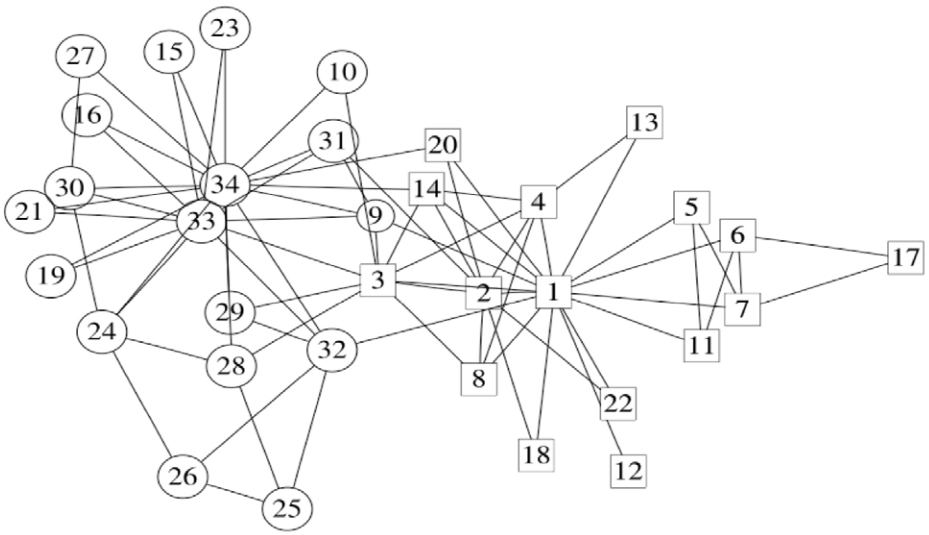

图 4　Zachary 空手道俱乐部 GN 算法划分图

而真实世界中的社交网络往往是重叠复杂的关系,同一个节点可能属于多个社区[14],因此传统社区划分方法不能满足需要。而文中采用的基于边聚类的重叠社区算法,能够有效地将网络划分为相互重叠的社区。

如图 5 所示,将网络图划分为两个重叠社区,例如节点 2,3,8,14,20 同时属于两个社区。

4　结束语

从文中算法得到的实验结果能够看出,文中提出的基于边聚类的社区算法能够有效地发现重叠社区,并且发现的社区结构比传统的社区发现算法发现社区结构的效果更加有效、准确,更能说明现实世界中的情况。

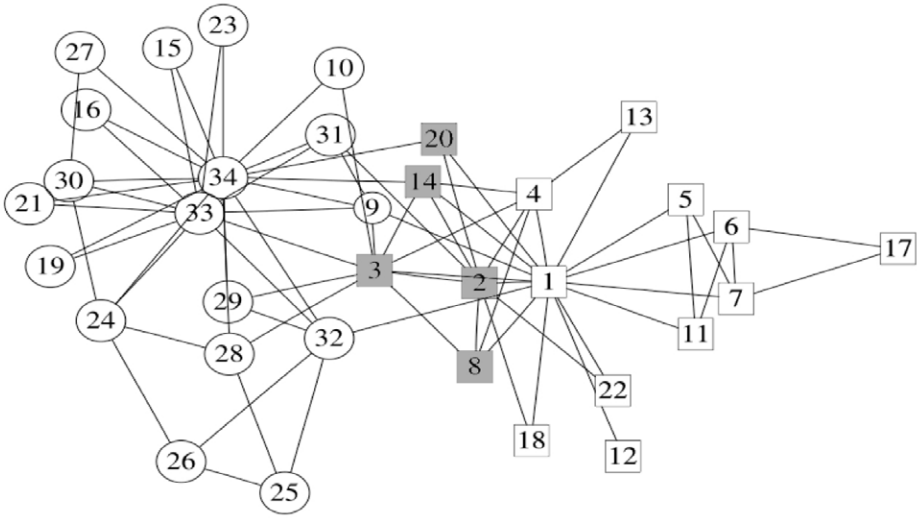

图 5 Zachary 空手道俱乐部基于边聚类算法划分图

另外,文中采用在 Hadoop 分布式框架下设计该算法,能够运用分布式框架的运算优势,在大规模图处理上实现对重叠社区的划分。接下来的工作将致力于大规模图上的重叠社区发现优化处理。

参考文献:

[1] Kernighan B W, Lin S. An efficient heuristic procedure for partitioning graphs [J]. Bell System Technical Journal, 1972, 49(2): 291 – 307.

[2] Pothen A, Simon H D, Liou K P. Partitioning sparse matrices with eigenvectors of graphs [J]. SIAM Journal on Matrix Analysis and Applications, 1990, 11(3): 430 – 452.

[3] Girvan M, Newman M E J. Community structure in social and biological networks [J]. Proceedings of the National Academy of Sciences, 2002, 99(12): 7821 – 7826.

[4] Newman M E J. Fast algorithm for detecting community structure in networks [J]. Physical Review E, 2004, 69(6): 066133.

[5] 骆志刚,丁 凡,蒋晓舟,等. 复杂网络社团发现算法研究新进展 [J]. 国防科技大学学报, 2011, 33(1): 47 – 52.

[6] Evans T S, Lambiotte R. Line graphs, link partitions, and overlapping communities [J]. Physical Review E, 2009, 80(1): 016105.

[7] Harary F. Graph theory [M]. Massachusetts: Addison – Wesley, 1972.

[8] Harary F, Norman R Z. Some properties of line digraphs [J]. Rendiconti del Circolo Matematico di Palermo, 1960, 9(2): 161 – 168.

[9] Zhou T, Ren J, Medo M, et al. Bipartite network projection and personal recommendation [J]. Physical Review E, 2007, 76(4): 046115.

[10] White T. Hadoop 权威指南 [M]. 曾大聃,周傲英,译. 北京: 清华大学出版社, 2010.

[11] 赵卫中,马慧芳,傅燕翔,等. 基于云计算平台 Hadoop 的并行 k – means 聚类算法设计研究 [J]. 计算机科学, 2011, 38(10): 166 – 168.

[12] 雒江涛,李晴川. 基于云存储的分组域监测系统 [J]. 重庆邮电大学学报: 自然科学版, 2012, 24(6): 675 – 681.

[13] 夏 英,杨选伦. 云环境中基于金字塔模型的影像数据存储方法 [J]. 重庆邮电大学学报: 自然科学版, 2012, 24(6): 669 – 674.

[14] 施 伟,傅鹤岗,张 程. 基于连边相似度的重叠社区发现算法研究 [J]. 计算机应用研究, 2013, 30(1): 221 – 223.

[15] Shang Mingsheng, Chen Duanbing, Tao Zhou. Detecting overlapping communities based on community cores in complex networks [J]. Chinese Physics Letters, 2010, 27(5): 058901.

[16] 武志昊,林友芳,田盛丰,等. 高度重叠社区的社区合并优化算法 [J]. 北京交通大学学报: 自然科学版, 2011, 35(3): 116 – 122.