第37卷 第14期 计 算 机 工 程 2011年7月
Vol.37 No.14 Computer Engineering July 2011

·软件技术与数据库· 文章编号：1000—3428(2011)14—0041—03 文献标识码：A 中图分类号：TP391.4

一种基于粗糙集的社区结构发现算法

朱文强[1]，伏玉琛[1,2]

(1. 苏州大学计算机科学与技术学院，江苏 苏州 215006；2. 江苏省现代企业信息化应用支撑软件工程技术研发中心，江苏 苏州，215104)

摘 要：提出一种基于粗糙集的社区结构发现算法。将信息中心度作为衡量节点之间关联度的标准，在处理社区间边界节点时引入粗糙集中的上下近似集概念。将网络中的各个节点划分到社区中，从而将复杂网络划分成 k 个社区，k 值由算法自动选定，并通过模块度确定理想的社区结构。在 Zachary Karate Club 模型和 College Football Network 模型上进行验证，实验结果表明，该算法的准确率较高。

关键词：社区结构；节点关联度；粗糙集；上近似集；下近似集

Community Structure Detection Algorithm Based on Rough Set

ZHU Wen-qiang[1], FU Yu-chen[1,2]

(1. School of Computer Science & Technology, Soochow University Suzhou, 215006, China; 2. Jiangsu Province Support Software Engineering R&D Center for Modern Information Technology Application in Enterprise, Suzhou 215104, China)

【Abstract】This paper proposes a new detection algorithm based on rough set. It uses information centrality as a measure of correlation between nodes. While dealing with the boundary nodes between communities, it uses upper and lower approximations subsets so as to better simulate the real world, then it clusters nodes to certain community and identify the network to k communities, identifies the ideally community structure according to modularity, besides the k value need not to be prior given. The algorithm is tested on two network dataset named Zachary Karate Club and College Football. and experimental result shows it has high accuracy rate.

【Key words】community structure; node relevance degree; rough set; upper approximations; lower approximations

DOI: 10.3969/j.issn.1000-3428.2011.14.012

粗糙集提供了一套处理不确定性问题的理论，本文将其应用在社区集合的聚类中，通过在聚类时使用上近似集合和下近似集合来表示属于同一社区的集合，并以节点关联度作为聚类距离，提出一种新的社区划分算法。

1 相关社区发现算法

Girvan 和 Newman 于 2002 年提出一种基于层次聚类的分裂算法[1]，称为 Girvan Newman 算法，其基本思想是不断从网络中移除介数最大的边，边的介数是指通过该边的最短路径的数目。由于同一社区内的节点对介数较小，而处在不同社区的节点对介数较大，因此可以比较好地划分社区。GN 算法准确率较高，但算法复杂度较大。

Kernighan-Liu 算法[2]基于贪婪算法原理将网络划分成 2 个大小已知的社区。其基本思想是为网络的划分引入增益函数 Q，表示 2 个社区内部的边数减去 2 个社区间的边数，然后寻找使 Q 值最大的划分方法。其算法复杂度约为 $O(n^2)$，其中，n 为网络节点数目。该算法的主要问题在于要求已知网络 2 个社区的大小。

谱平分法[3]是通过分析网络的拉普拉斯算子(Laplacian)的特征向量完成社区发现，在拉普拉斯矩阵的不为 0 的特征值所对应的特征向量中，同一个社区内的节点所对应的元素是近似相等的。当网络的确是分成 2 个社区时，用谱平分法可得到较好的效果，但其缺点在于每一次分割必须把网络分解成 2 个部分，通过反复调用算法来完成多社区划分，如果不能已知网络的社区数目，算法将很难达到满意效果。

文献[4]提出的 M-Chameleon 聚类算法基于结构等价和模块度的概念，而且无需人工选择 K 值，聚类沿结构相似度从小到大的顺序进行合并，并由模块度决定算法的终止点，但该算法在适用范围和效率上还有待提高。

2 基于粗糙集的网络社区结构发现

2.1 粗糙集

粗糙集[5]是一种处理不确定性知识的工具，下文给出与本文相关的部分粗糙集定义。

定义 1 设 U 表示研究对象组成的非空有限集合，称为论域。R 是 U 上等价关系的族集。近似空间(approximate space)定义为一个关系系统(或二元组)，用 $A=(U,R)$ 表示。

定义 2 等价关系的族集 R 将 U 划分为一系列不连续的子集，而这些子集使用 $U/R=E_1,E_2,\mathrm{L}\ ,E_n$ 来表示，E_1 是 R 的一个等价类，如果 2 个对象 $u,v\subseteq U/R$ 属于同一个等价类 $E\subseteq U/R$，则称 u 和 v 为不可区分的。

定义 3 对任何集合 $X\subseteq U$，X 的下近似集为 $\underline{A}(X)=\mathrm{U}\{Y\in U/R:Y\in X\}$，表示确定属于 X 的对象集合；X 的上近似集为 $\overline{A}(X)=\mathrm{U}\{Y\in U/R:Y\mathrm{I}\ X\neq\phi\}$，表示可能属于 X 的对象的集合。

设一个划分分类方案为 $U/R=X_1,X_2,\mathrm{L}\ ,X_n$，由于缺乏足够知识，因此不能准确定义 $X_i(1\leqslant i\leqslant n)$，但可使用现有的信息定义 X_i 的下近似集 $\underline{A}(X)$ 和 X_i 的上近似集 $\overline{A}(X)$，以此来描述 X_i。X_i 的下近似集 $\underline{A}(X)$ 和上近似集 $\overline{A}(X)$ 满足以下性质：

基金项目：国家自然科学基金资助项目(60873116)；江苏省现代企业信息化应用支撑软件工程技术研发中心开放基金资助项目(SX200902)

作者简介：朱文强(1985－)，男，硕士研究生，主研方向：数据挖掘，Web 文本挖掘；伏玉琛，副教授

收稿日期：2010-12-24 **E-mail：**20084227162007@suda.edu.cn

性质 1 $\phi \subseteq \underline{A}(X_i) \subseteq \overline{A}(X_i) \subseteq U$；

性质 2 $\underline{A}(X_i) \cap \underline{A}(X_j) = \phi, i \neq j$；

性质 3 $\underline{A}(X_i) \cap \overline{A}(X_j) = \phi, i \neq j$；

性质 4 一个对象 v 不属于任何一个下近似集$\Leftrightarrow v$ 至少属于 2 个上近似集。

性质 1 说明如果一个对象属于一个类 X_i 的下近似，则它必然属于 X_i 的上近似；性质 2 说明一个对象 $u_k \in U$ 至多为一个类 X_i 的下近似。

2.2 节点关联度

网络中 2 个相邻节点之间的关联度是由它们所共享的边来决定的。2 个相邻节点之间共享的边的信息中心度越小，它们不是社区间传输信息的路径的可能性就越大，则它们属于同一个社区的概率就越大，它们之间的连接就越紧密，关联度就越高。

文献[6]提出了一种基于信息中心度的社区探测算法。假设网络 G 有 n 个节点和 m 条边。为衡量节点传输信息的有效性，引入网络效率(NE)的概念。假设网络中 2 个节点之间的信息总是沿着最短路径传播的，则节点 i 和节点 j 之间的信息传输率 ε_{ij} 为它们之间最短路径长度 d_{ij} 的倒数(如果节点 i 和 j 之间不存在路径，则最短路径长度为∞，$\varepsilon_{ij}=0$)。整个网络 G 的信息有效率定义为各节点对的信息传输有效率的平均值 $NE[G]$，即：

$$NE[G]=\sum_{i\neq j\in G}\varepsilon_{ij}/[n(n-1)]=1/[n(n-1)]\sum_{i\neq j\in G}1/d_{ij}$$

其中，边 ε_{ij} 的信息中心度 $C_{e_{ij}}$ 定义为移除该边后整个网络的信息有效率的减少的相对量，即：

$$C_{e_{ij}}=\Delta NE/NE=(NE[G]-NE[G_{e_{ij}}])/NE[G]$$

通过分析可以看出，社区间的边的信息中心度比社区内部边的信息中心度大。显然节点 i 和节点 j 之间的边的信息中心度越小，它们的关联度就越大，属于同一个社区的概率就越大。因此，定义节点 v_i、节点 v_j 的节点关联度为：

$$nodeLink(v_i,v_j)=1-C_{e_{ij}}$$

其中，$e_{ij}\in E$，E 为网络的边的集合；$C_{e_{ij}}$ 为边 e_{ij} 的信息中心度。

在本文提出的粗糙集社区结构发现算法中，社区划分时不再是边界确定的社区，每个社区由下近似集和上近似集组成，下近似集包含确定属于该社区的对象，而上近似集中包含的对象，除去包含在某个下近似集中的对象外，也包含某些上近似集中的对象。

定义节点 v_i 和社区 c_j 的归属关联度为：

$$Link(v_i,c_j)=e_{\text{low}}\times\frac{1}{N_{\text{low}}}\sum_{x\in w_{\text{low}}}nodeLink(v_i,x)+e_{\text{up}}\times\frac{1}{N_{\text{up}}}\sum_{x\in w_{\text{up}}}nodeLink(v_i,x)$$

$$j=1,2,\text{L},k,e_{\text{low}}+e_{\text{up}}=1$$

其中，w_{low}、w_{up} 分别表示第 j 个社区的下近似集和上近似集；e_{low}、e_{up} 分别表示第 j 个社区的下近似集和上近似集在求关联度时的权重，一般 $e_{\text{low}}>e_{\text{up}}$；$N_{\text{low}}$、$N_{\text{up}}$ 分别表示第 j 个社区的下近似集和上近似集的对象数量。

2.3 模块度

本文引入模块度(Q)的概念。将网络划分为 k 个社区。定义一个 $k\times k$ 维的矩阵 $\boldsymbol{e}=\{e_{ij}\}$，其中元素 e_{ij} 表示网络中连接两个不同社区 i 和 j 的节点的边在所有边中所占的比例。矩阵中对角线上各元素之和为 $Tre=\sum_i e_{ii}$（e_{ii} 表示网络中连接社区 i 内部各节点的边在所有边的数目中所占比例)，定义每行(或者列)中各元素之和为 $a_i=\sum_j e_{ij}$，它表示与第 i 个社区中的节点相连的边在所有边中所占的比例。用下式来定义模块性的衡量标准：

$$Q=\sum_i(e_{ii}-a_i^2)=Tre-\|\boldsymbol{e}^2\|$$

其中，$\|\boldsymbol{x}\|$ 表示矩阵 $\boldsymbol{x}$ 中所有的元素之和。该式的意义是：网络中连接 2 个同种类型的节点的边(即社区内部边)的比例减去在同样的社区结构下任意连接这 2 个节点的边的比例的期望值。

2.4 算法描述

确定一个对象属于一个社区的上近似集还是下近似集的方法如下：设 x 为待聚类的点，c_i 为目标社区，$d(x,c_i)$ 表示 x 与 c_i 中所有节点的节点关联度平均值。令 $d(x,c_i)$ 表示为 x 与 c_i 社区的节点平均关联度最大，即 $d(x,c_i)=\max_{j\in[1,k]}d(x,c_j)$，再设 $d=d(x,c_i)-d(x,c_j),1\leqslant i,j\leqslant k$，且$i\neq j$，给定上近似集和下近似集之间的阈值为 threshold，则有集合：

$$Z=\{j:d=d(x,c_i)-d(x,c_j)\leqslant threshold,i\neq j\}$$

规定 1 若$Z\neq\phi,x\in\overline{A}(w_i)$，则$x\in\overline{A}(w_j),\exists j\in Z$，且$x\notin\forall\underline{A}(w)$，这样就保证了性质 4 必然成立，即 x 不属于任何一个集合的下近似；

规定 2 若 $Z=\phi,x\in\overline{A}(w_i)$，为满足性质 1~性质 3，则 $x\in\underline{A}(w_i)$。通过以上 2 条规定可确定一个对象是属于上近似集还是下近似集。因此，可将节点对象聚类到粗糙集中，并且由 threshold 控制上近似和下近似的范围。

社区结构发现算法如下：

输入 网络的邻接矩阵

输出 网络的社团结构

(1)设 $c_1=\phi$（c_1 为第 1 个社区的集合），$c_2=\phi$（c_2 为第 2 个社区的集合），$k=2$。此外，设 $V_1=\{c_1,c_2\}$（已被聚类的节点集合），$V_2=V-V_1$（未被聚类的点)。

(2)选择节点集 V_2 中度最大的节点 v_1 作为第一个聚类的中心：$c_1=c_1\cup\{v_1\}$。

(3)观察 V_2 中剩余的节点，找到与 V_1 集合中平均关联度最低的点 v_j，如多个则可随意选择，令 $c_k=c_k\cup\{v_j\}$。

(4)如果 $|V_1|\neq k$，$k=k+1$，转步骤(3)；否则，转步骤(5)。

(5)根据节点与社区归属关联度公式和规定 1 及规定 2 将数据对象与社区中所有节点的平均关联度确定上近似集和下近似集合。

(6)如果 $V_2\neq\phi$，计算节点 $v_j(j=1,2,\text{L},|V_2|)$ 与哪个聚类的关联度最大，它就属于哪个聚类。一个聚类就是一个社区的划分结果。

(7)计算当前社区的模块度。如果 $Q_k\geqslant Q_{k-1}$，则 $k=k+1$，转步骤(3)；否则结束。

3 实验与结果分析

本文使用了 2 个经典模型 Zachary Karate Club 和 College Football Network 进行验证。

3.1 Zachary Karate Club 模型

Zachary Karate Club 网络是一个经典的问题，很多分析

复杂网络社区结构算法中都用到了该案例。用本文的算法去分析 Zachary Karate Club 网络，当把网络聚类成如图 1 所示的 2 个社区时，社区的模块度 Q 最大。聚类结果较好，只有 3 号节点出现划分错误。聚类的正确率达到了 97%。

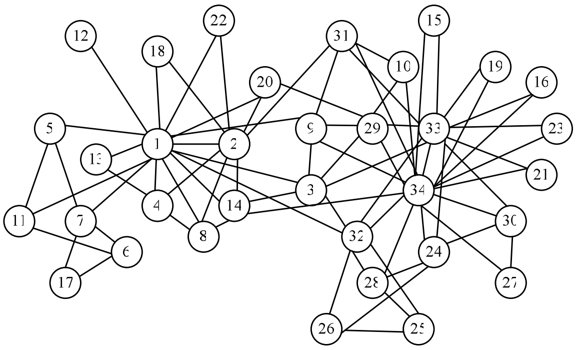

图 1　本文算法对 Zachary Karate Club 的划分结果

3.2　College Football Network 模型

College Football Network 模型是美国大学生足球联赛抽出来的一个复杂网络模型。足球联赛中有若干支球队，网络的节点代表一支球队，2 个节点之间的边代表 2 支球队之间进行过一场比赛。联赛中存在若干的联盟，每个球队都属于其中一个联盟。联盟内部的球队之间进行的比赛次数多于联盟之间的球队之间进行的比赛次数。存在 115 支球队(节点)及 616 场比赛(边)，包含了 12 个联盟。通过本文算法解决球队联盟划分的问题，当划分为 12 个社团时 Q 值达到最大。本文中的算法划分正确的节点为 96 个，正确率约为 83%，而 Newman 快速社区结构检测算法的准确率为 78%。

4　结束语

复杂网络社区结构发现是一个具有挑战性的研究领域，本文尝试使用粗糙集的理论来优化社区结构的划分，提出的算法将粗糙集的概念引入社区内节点聚类当中，建立了基于粗糙集的社区发现算法。与其他算法相比，即使在未知社区数量和社区内节点的数量的情况下，该算法的准确率也较高。重叠社区在网络社区结构发现中已经越来越受到重视，如何应用粗糙集理论对复杂网络社区结构进行重叠社区发现是下一步工作的重点。此外，本文中所针对的是无向无权重的简单图，在算法中考虑应对有向有权重的网络图的发现也是未来的研究方向。

参考文献

[1] Pawlak Z. Rough Sets[J]. Journal of Information and Computer Sciences, 1982, 11(5): 145-172.

[2] Kernighan B W, Lin Shen. An Efficient Heuristic Procedure for Partitioning Graphs[J]. Bell System Technical Journal, 1970, 49(1): 291-307.

[3] Pothen A, Smon H, Liou K P. Partitioning Sparse Matrices with Eigenvectors of Graphs[J]. SIAM Journal on Matrix Analysis and Applications, 1990, 11(3): 430-452.

[4] 龙真真, 张　策, 刘飞裔, 等. 一种改进的 Chameleon 算法[J]. 计算机工程, 2009, 35(20): 189-191.

[5] Giran M, Newman M E J. Community Structure in Social and Biological Networks[C]//Proc. of National Academy of Science, 2002, 99(12): 7821-7826.

[6] Fortunato S, Latora V, Iorim M. A Method to Find Community Structures Based on Information Centrality[J]. Physical Review E, 2004, 70.

编辑　金胡考

(上接第 40 页)

(10)产生执行报告，表明测试用例是成功或者失败的。

(11)产生了 2 种类型的覆盖报告：1)代码覆盖率；2)事件交互覆盖率。

(12)开发人员查看报告，修改缺陷。

(13)GUIART 产生新的事件流图和组合树，辨识可用和不可用的测试用例，修复可以修复的测试用例。

(14)产生新测试用例和预言信息。

(15)使用覆盖报告，执行报告，对测试用例排序。

第(9)步~第(15)步在被测应用程序的开发过程中重复。

5　结束语

GUI 不同于传统软件的特性给 GUI 测试提出了新的挑战。本文在以往研究的基础上，提出了基于 EIG 的 GUI 自动化回归测试框架，并说明了基于该框架的测试过程。下一步的主要工作是研究如何提高 GUI 回归测试的有效性。

参考文献

[1] Memon A M. A Comprehensive Framework for Testing Graphical User Interfaces[D]. Pittsburgh, USA: University of Pittsburgh, 2001.

[2] Memon A M, Xie Qing. Using Transient/Persistent Errors to Develop Automated Test Oracles for Event-driven Software[C]//Proc. of the International Conference on Automated Software Engineering. [S. l.]: IEEE Press, 2004: 186-195.

[3] 周　娟, 蒋外文. 基于 Web 的自动化测试框架[J]. 计算机工程, 2009, 35(18): 65-66.

[4] Memon A M, Xie Qing. Studying the Fault-detection Effectiveness of GUI Test Cases for Rapidly Evolving Software[J]. IEEE Transactions on Software Engineering, 2005, 31(10): 884-896.

[5] White L, Almezen H. Generating Test Cases for GUI Responsibilities Using Complete Interaction Sequences[C]//Proc. of the International Symposium on Software Reliability Engineering. [S. l.]: IEEE Press, 2000: 110-121.

[6] Xie Qing, Memon A M. Rapid Crash Testing for Continuously Evolving GUI-based Software Applications[C]//Proc. of the International Conference on Software Maintenance. [S. l.]: IEEE Press, 2005: 473-482.

编辑　顾逸斐