一种新的基于粗糙集 K-均值的社区发现方法

张云雷* 吴 斌 刘 宇
(北京邮电大学智能通信软件与多媒体北京市重点实验室 北京 100876)

摘 要：针对许多社区发现方法将社区看作一个集合而无法描述社区模糊区域的问题，该文提出一种基于粗糙集理论的社区发现方法。该方法将社区看作两个集合，即社区的下近似集和上近似集，来刻画社区的模糊区域。该方法首先选择 K 个节点作为社区的中心节点，然后根据节点与社区中心之间的距离将节点关联到社区中心节点形成社区，接着重新计算社区的中心点及节点的社区标签，如此迭代直到收敛。通过公开数据集和仿真数据集验证了该方法在社区发现方面的可行性和有效性。

关键词：社交网络分析；社区发现；粗糙集；K-均值
中图分类号：TP399 **文献标识码**：A **文章编号**：1009-5896(2017)04-0770-08
DOI: 10.11999/JEIT160516

A Novel Community Detection Method Based on Rough Set K-Means

ZHANG Yunlei WU Bin LIU Yu
(*Beijing Key Laboratory of Intelligent Telecommunications Software and Multimedia*, *Beijing University of Posts and Telecommunications*, *Beijing* 100876, *China*)

Abstract: Due to many community detection approaches regarding a community as one set of nodes which can not depict the vagueness of the community. A method based on rough set is proposed, it considers community as a lower and an upper approximation set which could depict the vagueness of the community. The method selects K nodes as the central nodes, then assembles iteratively nodes to their closest central nodes to form communities, and calculates subsequently a new central node in each community, around which to gather nodes again until convergence. Experimental results on public and synthetic networks verify the feasibility and effectiveness of the proposed method.

Key words: Social network analysis; Community detection; Rough set; K-Means

1 引言

聚类算法在数据挖掘中是一类重要算法，能够将具有相同性质的对象聚在一起。网络数据的聚类[1,2]也是聚类研究中的一个重要问题。网络提供了一种表示对象之间关系的方法，其中节点表示对象，而连边表示节点之间的关系。网络中的节点可以根据结构组织为社区，即具有一定共同性质、角色或功能的一组节点，比如具有一致功能的蛋白质[3]、具有共性的社交群体[4]。网络数据的聚类又称之为社区发现，是网络数据分析中一个重要研究问题。

在文献[1,2]中总结了目前社区发现的方法，主要分为基于图划分的方法、基于目标优化的方法、基于遗传算法的方法、基于标签传播的方法和基于语义的方法。其中基于图划分的典型方法 K-L 算法[5]以优化最小割边代价将图中的节点划分到给定大小的子集中；基于目标优化的典型方法有 Modularity[6]和 Louvain[7]。Modularity 方法是一种基于特征矩阵的特征向量表示的模块度矩阵，通过谱方法进行社区发现的方法；Louvain 方法是一种改进的模块度优化方法，是迭代式两阶段的算法，第 1 阶段找出能将网络模块度提高的社区划分，第 2 阶段将第 1 阶段的社区看成一个节点并加权，重复第 1 阶段的过程，具有较低的时间复杂度 $O(m)$，其中 m 为边数；基于遗传算法的方法 GA-Net[8]使用基于局部图表示，将节点描述为基因和染色体，定义和优化社区评分，最后能够发现网络中密集连接的社区；基于标签传播的方法 LPA[9]中，每个节点从其邻居节点的标签中获取出现次数最多的标签作为自己的标签，当每个节点的标签稳定时，算法结

收稿日期：2016-05-23；改回日期：2016-09-23；网络出版：2016-12-02
*通信作者：张云雷 yunlei0518@126.com
基金项目：国家重点基础研究发展计划(2013CB329606)，北京市共建项目
Foundation Items: The National Key Basic Research Program of China (2013CB329606), The Special Fund for Beijing Common Construction Project

束；基于语义的方法[10]基于观测连接-域-主题来估计边的语义权重，生成语义连接权重，进而生成语义社交网络的社区；近几年也有一些其他社区发现方法，文献[11]提出一种基于边界节点识别的局部社区发现算法，文献[12]提出一种基于动态距离的快速社区发现算法。上述社区发现方法将社区看成一个包含节点成员的集合，无法区分社区的核心节点与边缘节点，即每个社区中的节点具有相等的角色或重要程度，也就无法刻画社区的模糊区域。模糊区域包含了“犹豫”节点，“犹豫”节点以不同程度的“距离”归属到单个或多个社区。文献[13]使用粗糙聚类方法实现社区发现，但是一个节点最多只能归属到两个社区的上近似集。文献[14]提出节点关联度的粗糙聚类方法，但是容易将孤立点作为社区的核心节点，导致产生冗余社区。

本文提出一种基于粗糙集理论的社区结构发现算法 CDRS(Community Detection based on Rough Set)，将社区定义为粗糙社区，粗糙社区包含了两个集合：社区的上近似集和社区的下近似集。其中社区的下近似集包含社区的核心节点，即与社区联系紧密的节点；社区的上近似集不仅包含与社区紧密联系的核心节点，也包含与社区联系不紧密的边缘节点；而社区的边界区域包含了属于社区上近似集而不属于社区下近似集的节点。通过社区的边界区域能够刻画社区的模糊区域。本文方法的主要思想是利用节点的网络结构计算出节点之间的结构相似性，利用节点之间的相似性构建节点画像，将节点从网络空间转换到欧式空间，通过节点的中心性选出社区的中心节点，通过定义的距离度量将节点关联到社区的近似集中，收敛后得到社区结果。本文的主要创新有：(1)提出了粗糙社区的概念，将社区看作两个集合，而不是传统的一个集合；(2)提出一种能发现粗糙社区的方法；(3)通过实验验证提出的方法一定程度上优于目前的方法。本文第 2 节提出和分析了基于粗糙集的社区发现方法；第 3 节利用公开数据集和人工数据集进行实验验证，结果表明该算法的有效性和合理性；最后给出结论。

2　基于粗糙集的社区发现方法(CDRS)

2.1 预备知识与定义

为了更好说明粗糙社区的概念，图 1 给出了示例网络以说明传统社区和粗糙社区的区别。在图 1(a)中有两个社区，每个社区描述为一个集合。在图 1(b)中，左边的社区包含两种类型节点：一类是与社区联系紧密的节点，包括{2,3,4}；另一类节点与社区联系不紧密的节点，包括{1}，节点 1 是社区的边缘节点。同时在图 1(b)中，右边社区也包含了两种类型节点：一类是与社区联系紧密的节点，包括{6,7,8,9}；另一类节点与社区联系不紧密的节点，包括节点{5,10}，节点 5 和 10 是边缘节点，其中节点 5 是两个社区相连接的节点。

本文探讨简单无向无权图的社区发现问题。给定一个无向无权图 $\boldsymbol{G}=(\boldsymbol{V},\boldsymbol{E})$，$\boldsymbol{V}$ 是节点集合，$\boldsymbol{E}$ 是节点间连接集合。节点结构的定义为节点邻居及自身。

定义 1　节点结构：设 $v\in\boldsymbol{V}$，则节点 v 的结构定义为邻居节点及自身，记为 $\mathbf{NB}(v)$：

$$\mathbf{NB}(v)=\{w\in\boldsymbol{V}\mid(w,v)\in\boldsymbol{E}\}\cup\{v\} \tag{1}$$

定义 2　结构相似性：设 $v,w\in\boldsymbol{V}$，则节点 v,w 之间的结构相似性通过两节点的共同邻居个数与两节点邻居个数乘积的算术平方根的比值，记为 $s(v,w)$：

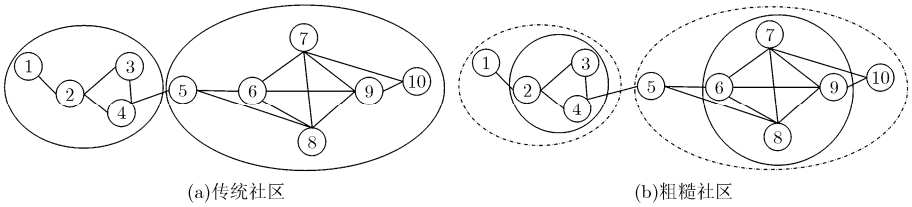
$$s(v,w)=\frac{|\mathbf{NB}(v)\cap\mathbf{NB}(w)|}{\sqrt{|\mathbf{NB}(v)|\cdot|\mathbf{NB}(w)|}} \tag{2}$$

此处，通过两个节点的共同邻居定义其结构相似性。如果节点 v 和 w 直接相连，两者之间共同邻居越多，其结构相似性 $s(v,w)$ 就越大。如果两者具有相似的拓扑结构，那么他们也许具有相似的功能。网络拓扑上的相似性决定了两个节点之间的相似的程度。

定义 3　节点画像：设 $v,w_i\in\boldsymbol{V},i=1,2,\cdots,N$，将节点 v 定义为一个向量，该向量由节点 v 与网络其他节点的结构相似性构成，记为 $\mathbf{NP}(v)$：

$$\mathbf{NP}(v)=(s_1,s_2\cdots s_i\cdots s_N) \tag{3}$$

其中，$N=|\boldsymbol{V}|$ 是网络中节点的个数，$s_i=\delta(v,w_i)*s(v,w_i)$，如果 $(v,w_i)\in\boldsymbol{E}$，那么 $\delta(v,w_i)=1$，否则 $\delta(v,w_i)=0$。需要注意，如果节点 v 的度比较小，则节点画像是稀疏向量，否则是稠密向量。



(a)传统社区　　(b)粗糙社区

图 1 传统社区与粗糙社区区别的示例网络

定义 4 等价类：设 $\boldsymbol{V}$ 是非空集合，$\boldsymbol{R}$ 是等价关系。对于 $v\in\boldsymbol{V}$，节点 v 的 $\boldsymbol{R}$ 等价类定义为[15]：

$$[v]_{\boldsymbol{R}}=\{x\mid x\in\boldsymbol{V},(v,x)\in\boldsymbol{R}\} \tag{4}$$

$$\boldsymbol{V}=\bigcup_{i=1}^{i=K}[v_i]_{\boldsymbol{R}} \tag{5}$$

$$\boldsymbol{V}/\boldsymbol{R}=\{[v_1]_{\boldsymbol{R}},[v_2]_{\boldsymbol{R}},\cdots,[v_K]_{\boldsymbol{R}}\} \tag{6}$$

其中，K 是等价类的个数，并且当 $i\neq j$ 时，$[v_i]_{\boldsymbol{R}}\cap[v_j]_{\boldsymbol{R}}=\boldsymbol{\Phi}$，$\boldsymbol{V}/\boldsymbol{R}$ 是集合 $\boldsymbol{V}$ 的等价划分。

定义 5 上、下近似集与边界区域：对于集合 $\boldsymbol{C}\subset\boldsymbol{V}$，$\boldsymbol{C}$ 的上近似集、下近似集和边界区域分别定义为 $\boldsymbol{U}(\boldsymbol{C})$，$\boldsymbol{L}(\boldsymbol{C})$ 和 $\boldsymbol{B}(\boldsymbol{C})$，形式如式(7)式~(9)[15]：

$$\boldsymbol{U}(\boldsymbol{C})=\{v\mid v\in\boldsymbol{V},[v]_{\boldsymbol{R}}\cap\boldsymbol{C}\neq\boldsymbol{\Phi}\} \tag{7}$$

$$\boldsymbol{L}(\boldsymbol{C})=\{v\mid v\in\boldsymbol{V},[v]_{\boldsymbol{R}}\subset\boldsymbol{C}\} \tag{8}$$

$$\boldsymbol{B}(\boldsymbol{C})=\boldsymbol{U}(\boldsymbol{C})-\boldsymbol{L}(\boldsymbol{C}) \tag{9}$$

显而易见，$\boldsymbol{\Phi}\subseteq\boldsymbol{L}(\boldsymbol{C})\subseteq\boldsymbol{C}\subseteq\boldsymbol{U}(\boldsymbol{C})$，$\boldsymbol{L}(\boldsymbol{C})$ 包含了确定属于社区 $\boldsymbol{C}$ 的节点，$\boldsymbol{U}(\boldsymbol{C})$ 包含了确定及可能属于社区 $\boldsymbol{C}$ 的节点，$\boldsymbol{B}(\boldsymbol{C})$ 包含了可能属于社区 $\boldsymbol{C}$ 的节点，用来刻画社区的模糊区域。节点 v 与社区 $\boldsymbol{C}$ 的近似集之间满足如下 3 条性质[16]：(1)节点 v 最多只能属于一个社区的下近似集；(2)如果节点 v 属于社区 $\boldsymbol{C}$ 的下近似集，那么节点 v 一定属于社区 $\boldsymbol{C}$ 的上近似集；(3)如果节点 v 属于两个及以上社区的上近似集，那么节点 v 不属于任何社区的下近似集。

定义 6 社区中心：对于第 i 个社区 $\boldsymbol{C}^i$，其中心可以是一个网络中的节点也可以是一个虚拟节点，社区中心是一个向量，由该社区中的节点画像 $\mathbf{NP}(v)=(s_1,s_2,\cdots,s_i,\cdots,s_N)$ 计算得来，记为 $\mathbf{CC}(\boldsymbol{C}^i)=(\boldsymbol{C}_1^i,\boldsymbol{C}_2^i,\cdots,\boldsymbol{C}_j^i,\cdots,\boldsymbol{C}_N^i)$。传统方法计算 $\mathbf{CC}(\boldsymbol{C}^i)$ 的分量 $\boldsymbol{C}_j^i$ 公式如式(10)：

$$\boldsymbol{C}_j^i=\frac{1}{N_i}\sum_{v\in\boldsymbol{C}^i}s_j \tag{10}$$

其中，N_i 是社区 $\boldsymbol{C}^i$ 包含节点的个数，s_j 是节点 v 画像的第 j 个分量。在引入了社区 $\boldsymbol{C}^i$ 的上下近似集之后，解决了社区 $\boldsymbol{C}^i$ 模糊区域描述的问题。则社区 $\boldsymbol{C}^i$ 可以使用两个近似集来表示，即 $(\boldsymbol{L}(\boldsymbol{C}^i),\boldsymbol{U}(\boldsymbol{C}^i))$。本文中，在计算社区中心 $\mathbf{CC}(\boldsymbol{C}^i)$ 的分量 $\boldsymbol{C}_j^i$ 时，不仅考虑了社区 $\boldsymbol{C}^i$ 的下近似集 $\boldsymbol{L}(\boldsymbol{C}^i)$ 中的节点，同时也考虑了边界区域 $\boldsymbol{B}(\boldsymbol{C})=\boldsymbol{U}(\boldsymbol{C})-\boldsymbol{L}(\boldsymbol{C})$ 中的节点，则改进的计算社区中心向量 $\mathbf{CC}(\boldsymbol{C}^i)$ 的分量 $\boldsymbol{C}_j^i$ 的公式为

$$\boldsymbol{C}_j^i=\begin{cases}\alpha\times\dfrac{\sum\limits_{v\in\boldsymbol{L}(\boldsymbol{C}^i)}s_j}{\left|\boldsymbol{L}\left(\boldsymbol{C}^i\right)\right|}+\beta\times\dfrac{\sum\limits_{v\in\boldsymbol{B}(\boldsymbol{C}^i)}s_j}{\left|\boldsymbol{B}\left(\boldsymbol{C}^i\right)\right|}, & \boldsymbol{B}(\boldsymbol{C}^i)\neq\boldsymbol{\Phi}\\ \dfrac{\sum\limits_{v\in\boldsymbol{L}(\boldsymbol{C}^i)}s_j}{|\boldsymbol{L}\left(\boldsymbol{C}^i\right)|}, & \text{其他}\end{cases} \tag{11}$$

式(11)中，$|\boldsymbol{L}(\boldsymbol{C}^i)|$ 表示社区 $\boldsymbol{C}^i$ 下近似集包含节点的个数，$|\boldsymbol{B}(\boldsymbol{C}^i)|$ 表示社区 $\boldsymbol{C}^i$ 边界区域包含节点的个数。s_j 是节点 v 画像的第 j 个分量。α 是下近似集的加权参数，β 是边界区域的加权参数，且 $\alpha+\beta=1$。如果 $\beta=0$，则边界区域中的节点将不参与社区中心向量的计算。

定义 7 节点与社区的距离：节点与社区的距离定义为节点向量与社区中心向量的距离，则节点向量 $\mathbf{NP}(v)=(s_1,s_2,\cdots,s_i,\cdots,s_N)$ 与社区中心向量 $\mathbf{CC}(\boldsymbol{C}^i)=(\boldsymbol{C}_1^i,\boldsymbol{C}_1^i,\cdots,\boldsymbol{C}_j^i,\cdots,\boldsymbol{C}_N^i)$ 之间的距离计算公式如式(12)：

$$d(v,\boldsymbol{C}^i)=\sqrt{\frac{\sum_{j=1}^{N}\left(s_j-\boldsymbol{C}_j^i\right)^2}{N}} \tag{12}$$

其中，$N=|\boldsymbol{V}|$ 是网络中节点的个数。

使用 $d(v,\boldsymbol{C}^i)/d(v,\boldsymbol{C}^j)$ 比值决定节点 v 与两个社区之间的关系，规则如下[16]：

(1)为找到社区的上近似集，引入参数 $\gamma\geq1$，不失一般性，设 $d(v,\boldsymbol{C}^i)/d(v,\boldsymbol{C}^j)\leq\gamma,1\leq i,j\leq K$ 并且 $i\neq j$，K 表示社区的个数。如果 $v\in\boldsymbol{B}(\boldsymbol{C}^i)$，那么 $v\in\boldsymbol{B}(\boldsymbol{C}^j)$。进一步说，节点 v 不属于任何社区的下近似集，此规则满足了节点与社区近似集之间关系的性质(3)。

(2)否则 $v\in\boldsymbol{L}(\boldsymbol{C}^i)$，其中 $1\leq i\leq K$。同时根据性质(2)有 $v\in\boldsymbol{U}(\boldsymbol{C}^i)$，根据性质(1)有节点 v 不会属于其他社区的下近似集。

2.2 基于粗糙集的社区发现算法(CDRS)

2.2.1 CDRS 算法框架 社区的中心节点是社区中最重要的成员，并且每个社区由一个最重要的中心节点及与其相关联的节点组成。CDRS 的基本思想就是首先找到每个社区的最重要的中心节点，即 K 个中心点。然后根据网络中其他节点与 K 个中心点之间的距离，将节点关联到 K 个社区的上下近似集中。

表 1 给出了 CDRS 的主要步骤。首先选择 K 个社区的初始中心节点，然后在将节点关联到社区的近似集与计算新的中心节点之间迭代。根据网络的边集合，计算直接相连节点之间的相似度；接着选出 K 个社区的中心节点，将网络中其他节点与中心节点相关联形成社区的上下近似集，然后根据社区的上下近似集中的节点重新计算社区中心节点，如此迭代，直到社区中心不再发生变化。

2.2.2 初始化方法 初始社区中心节点选择是关键。选择正确的中心节点将会加速算法的收敛速度，而选择错误的中心节点必然导致额外的迭代次数，甚至陷入局部最优。为了能够选择正确的社区中心节

表 1 算法 1 —— CDRS 算法

输入：社交网络 $\boldsymbol{G}=(\boldsymbol{V},\boldsymbol{E})$；社区个数 K
输出：社区集合 $\boldsymbol{C}=\{\boldsymbol{C}_1,\boldsymbol{C}_2,\cdots,\boldsymbol{C}_K\}$
(1) **for** 所有的边 $(u,v)\in\boldsymbol{E}$ **do**
(2)　根据定义 3 计算节点 u 与 v 之间的结构相似度
(3) **end for**
(4) 根据算法 2 产生 K 个社区的中心节点
(5) **while** 当前社区中心存在与上次迭代产生的社区中心节点不同的中心节点 **do**
(6) //发现社区
(7) **for** 对 $\boldsymbol{V}$ 中每个节点 v **do**
(8)　　使用算法 3 将节点 v 赋值到对应社区的近似集中
(9) **end for**
(10) //更新社区的中心节点向量
(11) 使用算法 4 根据节点与社区近似集的关系生成社区集合 $\boldsymbol{C}$ 并重新计算社区的中心节点向量
(12) **end while**
(13) **return** $\boldsymbol{C}$

点，使用启发式规则选择社区的初始中心节点，按照节点的中心性选择社区中心节点。节点的中心性反映了节点在网络中重要程度。有许多衡量节点中心性的方法，比如度中心性、介数中心性、接近中心性等等。通过实验，选择度中心性作为本文的中心性度量方法，因为度中心性容易计算并且能得到不错的结果。如果直接按照中心性选择网络中前 K 个节点作为社区的中心节点，可能会导致错误的结果，因为前 K 个节点可能有的节点属于同一个社区。为了解决这个问题，应该选择不在相同社区的 K 个节点作为中心节点，在此，引入共同邻居个数参数 l，通过中心节点之间的共同邻居个数来判断候选中心节点是否在同一个社区，如果两个候选中心节点的 邻居个数大于 l，则认定两个节点属于同一社区，否则不在同一社区。该 K 个节点应该是对应社区中的具有最高中心性的节点，本文中采用贪心策略来选择前 K 个中心节点。主要思想是根据度中心性将节点从高到低排序，依次选取当前最高度中心性的节点作为候选中心节点，检查候选中心节点与当前已选出的中心节点之间的共同邻居数是否大于 l，如果共同邻居数大于 l，则放弃候选中心节点，否则将候选中心节点加入当前中心节点列表中。通过真实网络实验发现共同邻居个数 l 参数取值为 5 时，可以很好选择出各个社区的中心节点。算法详细步骤见表 2 的初始化方法。

2.2.3 将节点关联到社区近似集　社区的中心节点代表所在社区，将节点关联到社区近似集，就是通过节点与社区中心节点的关系判断节点与社区之间的关系。对于网络中的每个节点 v，通过计算节点 v 与各个社区中心节点 c 之间的距离，找出离节点 v 距离最近的中心节点与最小距离，然后通过节点 v 到其他中心节点的距离与最小距离的比值大小，根据节点与社区近似集的 3 条性质判定节点 v 与各个社区之间的关联关系。算法细节见表 3 的算法 3——节点关联社区近似集。

表 2 算法 2 —— 初始化方法

输入：社交网络 $\boldsymbol{G}=(\boldsymbol{V},\boldsymbol{E})$；社区个数 K；共同邻居个数 l
输出：中心节点列表 CNL；
(1) 将节点列表按照节点中心性从高到低排序
(2) 将具有最高中心性的节点添加到中心节点列表 CNL，并将该节点从有序节点列表中删除
(3) **while** 中心节点个数小于 K 时 **do**
(4)　//寻找下一个中心节点
(5)　从节点列表中选取中心性最高的节点 v 为候选中心节点
(6)　**if** 节点 v 与列表 CNL 中任意中心节点的共同邻居数小于 l **then**
(7)　　将节点 v 添加到中心节点列表 CNL 中
(8)　**end if**
(9)　**end while**
(10) **return** CNL

表 3 算法 3 —— 节点关联社区近似集

输入：社交网络 $\boldsymbol{G}=(\boldsymbol{V},\boldsymbol{E})$；中心节点列表 CNL；阈值 γ
输出：关联到社区近似集的节点集合 $\boldsymbol{V}$；
(1) 设 minDistance 表示节点与各个社区之间的最小距离，初始值为足够大
(2) 设 BCN 表示离节点最近社区的编号，初始值为-1
(3) **for** $\boldsymbol{V}$ 中的每个节点 v **do**
(4)　//寻找节点 v 与各个社区中心节点之间最小的距离
(5)　使用定义 7 计算节点 v 与 CNL 中各个中心节点之间的距离
(6)　找出离节点 v 距离最近的中心节点及距离，分别保存到 BCN 和 minDistance 中
(7) **end for**
(8) //将节点 v 关联到对应社区近似集
(9) **for** $\boldsymbol{V}$ 中的每个节点 v **do**
(10)　**for** 中心节点列表 CNL 中的每个中心节点 c **do**
(11)　　**if** $d(v,c)/d(v,\mathrm{BCN})\le\gamma$ **then**
(12)　　将节点 v 关联到中心节点 c 和 BCN 所对应的社区的上近似集中，分别为 $\boldsymbol{U}(\boldsymbol{C}^c)$ 与 $\boldsymbol{U}(\boldsymbol{C}^{\mathrm{BCN}})$
(13)　　**end if**
(14)　**end for**
(15)　**if** 节点 v 没有关联到两个及以上社区的上近似集中 **then**
(16)　　将节点 v 关联到中心节点 BCN 代表社区的上下近似集中，即 $\boldsymbol{L}(\boldsymbol{C}^{\mathrm{BCN}})$ 和 $\boldsymbol{U}(\boldsymbol{C}^{\mathrm{BCN}})$
(17)　**end if**
(18)　**end for**
(19)　**return** $\boldsymbol{V}$

2.2.4 生成社区与重新计算社区中心节点 经过关联节点到社区近似集的过程后，每个节点都具有了与社区关联的信息，可以根据关联社区信息生成社区与社区的中心节点。算法主要分为两步，首先根据节点关联社区的信息，将节点赋值到社区的上下近似集中，进而生成社区的近似集。然后根据定义 6 重新计算社区的中心节点。算法细节见表 4 的算法 4 —— 生成社区与重新计算社区中心节点。

表 4 算法 4——生成社区与重新计算社区中心节点

输入：算法 3 处理过的节点集合 $\boldsymbol{V}$；社区个数 K；中心节点列表 CNL；社区上近似集加权参数 β

输出：社区集合 $\boldsymbol{C}$；更新后的中心节点列表 CNL

(1) **for** 算法 3 处理过的节点集合 $\boldsymbol{V}$ 中的每个节点 v **do**

(4) //生成社区近似集

(5) 根据节点 v 的社区关联信息，将节点赋值到对应社区的近似集中，生成社区的近似集

(6) **end for**

(7) //更新社区中心节点

(8) **for** 中心节点列表 CNL 中的每个中心节点 c **do**

(9) 根据社区的近似集，按照定义 6 重新计算社区的中心节点向量并更新中心节点列表 CNL

(10) **end for**

(11) **return** $\boldsymbol{V}$ 和 CNL

2.3 复杂度分析

在算法 2 中，第 1 行将节点按照中心性排序所需时间复杂度为 $O(N\log_2 N)$，第 2~8 行选择 K 个中心节点的时间复杂度与中心节点的邻居个数成正比，该时间远小于其他时间，可以看作常量。在算法 3 中，第 3~7 行需要找出每个节点与 K 个中心节点之间的最小距离，时间复杂度为 $O(N)$，第 9~18 行将每个节点关联到相应社区近似集中的时间复杂度为 $O(KN)$。在算法 4 中根据每个节点的关联社区信息生成社区的近似集与社区中心节点向量，其时间复杂度为 $O(N)$。在算法 1 中，第 1~3 行计算相连节点之间的相似度时间复杂度为 $O(M)$，第 4 行就是算法 2 的时间复杂度，第 7~11 行生成社区近似集与更新社区中心节点的时间复杂度就是算法 3 和算法 4 的时间复杂度。因此，算法 1 的时间复杂度是 $O(M+N\log_2 N+tKN+tN)$，其中 M 是网络中边的个数，N 是节点个数，K 是社区个数，t 是迭代次数，一般情况下 $2\le t\le 10$。

3 实验分析

选择对比方法 通过与公认的较好的社区发现方法做对比来评估本文方法的性能。

Modularity[6]方法是目前基于模块度优化的最流行的社区发现方法。

MCL[17]是一种在生命科学中被广泛使用的、基于模拟随机流的图聚类算法。

Louvain[7]是基于模块度优化的著名算法，它能够发现网络中的层次结构并具有较低的时间复杂度。

LPA[9]是一种著名的通过标签传播发现社区的算法，具有较低的时间复杂度。

评价指标 通过将发现的社区结果与真实的网络划分相比较来评价方法的性能。本文采用聚类常用的评价指标 ARI (Adjusted Rand Index)[18]与 NMI (Normalized Mutual Information)[19]的组合指标，该值越高越好。

3.1 参数设置与实验环境

社区的个数是 CDRS 的一个主要参数，该值可由领域专家给定或者其他不需要此参数的社区发现方法中获取。然而许多情况下，其他不需要此参数的社区发现方法也不能正确获取社区的个数。本文中，设置 K 为网络的真实社区个数和其他方法发现社区个数。

阈值 γ 是 CDRS 的另一重要参数，γ 的值决定了社区的边界区域的大小。γ 值越大，社区的边界区域包含越多的节点。即随着 γ 的增大，社区包含了更多的不确定的节点，社区变的模糊。实验中，γ 值设置为 1.1。

对于共同邻居阈值 l 设置，如果两个节点的共同邻居数大于 l，那么该两个节点就是在同一个社区。如果 l 值过小，比如 2, CDRS 就不能找到较高中心性的节点作为社区的中心节点；如果 l 过大，CDRS 虽能找到较高中心性作为中心节点，但这些中心节点中可能存在属于同一社区的情况。本文中，通常将 l 设置为 5。在实验中，将社区的上下近似集同等看待，在计算社区中心节点时，对于上下近似集的加权参数 α 与 β 都设置为 0.5。

MCL[17]使用默认的膨胀系数(i=2.0)。所有的实验都是在 PC 上运行，配置为 3.6 GHz 的 CPU, 8 G 的内存、Windows10 操作系统。

3.2 人工合成网络实验

首先生成具有不同参数的人工网络来评价不同社区发现方法的性能。为公平起见，使人工网络更像真实网络，使用 LFR[20]人工合成网络工具，该工具可以灵活控制度分布和社区大小分布。混合参数 μ 代表节点与外部社区连接边的比例，该值越高，代表社区之间划分越不明显，通过增加 μ 值，生成

更复杂的人工网络。通过变化社区间的连边比例来评测社区发现方法性能的优劣。通过固定网络节点的平均度和社区大小，将 μ 的取值范围定为[0.1,0.8]来生成系列的人工网络，所有网络包含 2000 个节点，节点平均度为 20。随着混合参数 μ 的增加，4 种算法的 NMI 性能对比如图 2 所示。可以看到在 μ 小于 0.5 时，CDRS 方法和 Louvain 方法性能最好，随着 μ 增加，4 种方法的性能都有不同程度的下降，而 CDRS 相对来说比较稳定，而 MCL 方法和 LPA 方法对 μ 值比较敏感。

3.3 真实网络实验

通过公开真实网络评价各算法的性能，该真实网络数据均来自 Mark Newman 提供的网站[21]，真实网络的数据特征如表 5 所示。

在中心节点选择的策略中，本文分别以随机、度中心性、介数中心性和接近度中心性选择策略来选择中心点，其中随机选择中心点的实验结果是运行 5 次实验的平均值，实验结果如图 3 所示，可以看出度中心性选择策略取得了最好的效果，故本文采用度中心性为选择中心节点的策略。实验效果如图 4 所示，其中 CDRS 代表本文方法，M 代表 Modularity 算法，L 表示 Louvain 算法，MCL 代表 MCL 方法，LPA 表示 LPA 方法，图中采用 NMI 和 ARI 组合指标作为性能指标，横坐标中的数字表示社区个数。当 CDRS 的社区个数设置为真实社区个数时，CDRS 的评价结果获得最好的效果；当 CDRS 设置为对比方法发现的社区个数时，CDRS 的评价结果也都是优于大部分方法的评价结果。

3.3.1 空手道俱乐部网络　此网络来自于 Zachary 研究的一个著名的空手道俱乐部，网络中 34 个节点代表 34 个俱乐部成员，连边代表了成员之间的联系。在此研究中，观测了两年的 34 位成员之间的关系。期间因为管理者和教练的意见不一致，最终导致俱乐部分为两个小组，分别以管理者和教练为中心。

表 5 数据集特征统计表

数据集	节点数	边数	社区数	平均度	聚集系数
Zachary	35	78	2	4.588	0.571
Football	115	613	11	10.661	0.403
Polbooks	105	441	3	8.400	0.488

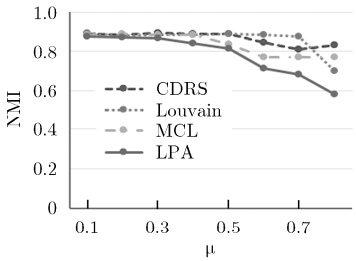


图 2　LFR 网络下 4 种算法 NMI 指标性能对比

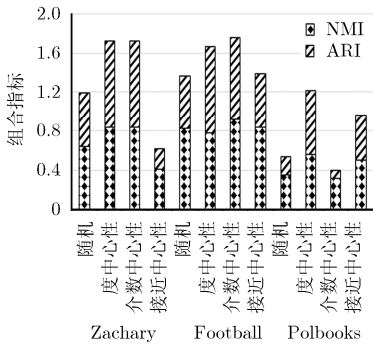
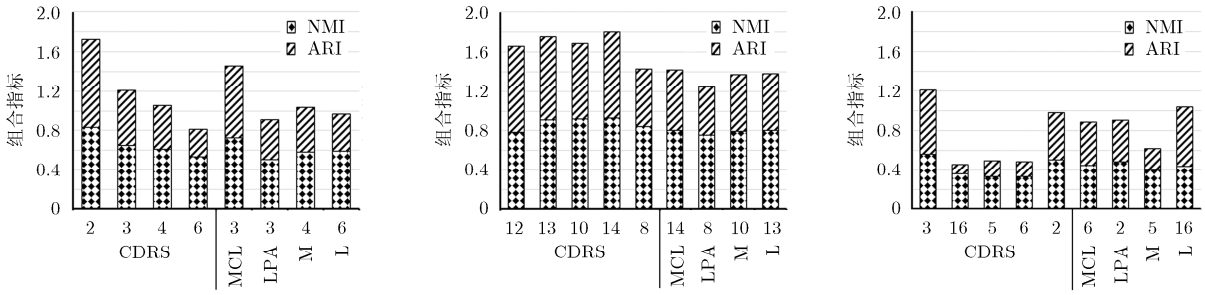


图 3 中心节点选择策略对比



(a)在Zachary-Karate上的对比　(b)在college football上的对比　(c)在polbooks上的对比

图 4 真实网络中 4 种算法组合指标性能对比

图 5 中，节点 1, 34 是 CDRS 发现的初始社区中心节点，事实上该两个节点正是对应社区的重要节点。当γ=1.1 时，CDRS 发现的社区效果如图 5 所示，节点 10 归属到左边社区；当γ=1.25 时，CDRS 将社区划分为上下近似集，其中节点 10 变为两个社区的上近似集中的元素，即将节点 10 关联到了两个社区的上近似集，节点 10 归属到两个社区。这是与传统社区区别之处。从图 5 中可以看出，节点 10 与两个社区都有连接，且连接紧密程度相近。如图 4(a)所示，当 CDRS 的社区个数为 2 时，获得最高的组合指标值，MCL 也达到了不错的效果，而 Modularity, Louvain 和 LPA 获得的组合指标值较低，不能较好发现社区。

3.3.2 美国大学足球网络 该网络是 2000 年时美国部分大学之间的秋季足球比赛，其中 115 个节点代表了 115 个队伍，613 条边代表了两个队伍之间的比赛。这些队伍被分为 12 个联盟，每个联盟大概包括 8-12 个队伍。如图 4(b)所示，CDRS 能够以较高的组合指标值发现美国大学足球网络的社区。MCL 的结果与 CDRS 相当，LPA 发现社区的效果最差。

3.3.3 美国政治书籍网络 该网络来自于 2004 年美国总统大选时发布的政治书籍，包含了 105 个节点，441 条边。节点代表亚马逊网站所卖出的政治书籍，节点之间的边代表有人同时购买了这两个节点所代表的书。这些政治书籍分为 3 类：自由党、中立、保守党。真实数据中，这些政治书籍被分为 3 类，其中两个类包含的书籍较多，一个类包含的书籍较少。CDRS 发现的社区大小的分布与真实情况一致。如图 4(c)所示，CDRS 的组合指标值都高于其他方法。

4 结束语

本文提出一种新的方法 CDRS 发现网络中隐含的社区结构。通过粗糙集的概念，CDRS 将社区描述为社区的上下近似集，如此能够刻画社区的模糊区域。通过人工网络和真实网络实验，CDRS 能够有效地发现社区。虽然 CDRS 需要社区的个数作为输入参数，但是有些情况能够获取社区的个数，CDRS 具有一定的价值。另外其他很多不需要社区个数作为参数的社区发现方法也是需要其他参数。通过对社区上近似集的进一步处理以发现重叠社区。CDRS 的时间复杂度较高，目前适合在中小型网络中发现结构社区，比如线下的社交网络等。在未来的工作中，优化算法使之应用于大规模网络是未来研究方向之一。

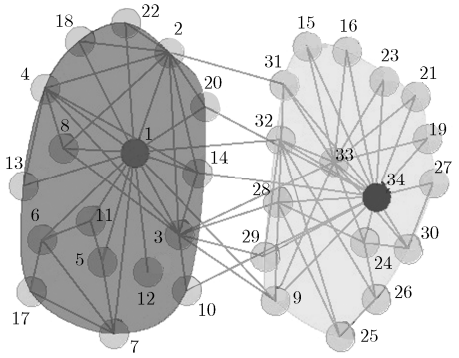


图 5 CDRS 发现空手道俱乐部网络社区示意图

参 考 文 献

[1] FORTUNATO S. Community detection in graphs[J]. *Physics Reports*, 2010, 486(3-5): 75–174. doi: 10.1016/j.physrep.2009.11.002.

[2] BEDI P and SHARMA C. Community detection in social networks[J]. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2016, 6(3): 115–135. doi: 10.1002/widm.1178.

[3] KROGAN N J, GERARD C, HAIYUAN Y, *et al*. Global landscape of protein complexes in the yeast saccharomyces cerevisiae[J]. *Nature*, 2006, 440(7084): 637–643.

[4] WASSERMAN S and FAUST K. Social Network Analysis Methods and Applications[M]. Cambridge, UK, Cambridge University Press, 2010, Chapter 7.

[5] KERNIGHAN B W and LIN S. An efficient heuristic procedure for partitioning graphs[J]. *Bell Labs Technical Journal*, 1970, 49(2): 291–307. doi: 10.1002/j.1538-7305.1970.tb01770.x.

[6] NEWMAN M E. Modularity and community structure in networks[J]. *Proceedings of the National Academy of Sciences*, 2006, 103(23): 8577–8582.

[7] BLONDEL V, GUILLAUME J, LAMBIOTTE R, *et al*. Fast unfolding of communities in large networks[J]. *Journal of Statistical Mechanics Theory & Experiment*, 2008, 30(2): 155–168.

[8] PIZZUTI C. GA-Net: a genetic algorithm for community detection in social networks[J]. *Parallel Problem Solving from Nature–PPSN X. Berlin Heidelberg: Springer-Verlag*, 2008, LNCS 5199:1081–1090. doi:10.1007/978-3-540-87700-4_107.

[9] USHA NANDINI R, RKA A, and SOUNDAR K. Near linear time algorithm to detect community structures in large-scale networks[J]. *Physical Review E Statistical Nonlinear & Soft Matter Physics*, 2007, 76(3): 036106.

[10] XIN Y, YANG J, and XIE ZQ. A semantic overlapping community detection algorithm based on field sampling[J].

Expert Systems with Applications, 2015, 42: 366–375. doi: 10.1016/j.eswa.2014.07.009.

[11] 刘阳, 季新生, 刘彩霞. 一种基于边界节点识别的复杂网络局部社区发现算法[J]. 电子与信息学报, 2014, 36(12): 2809–2815. doi: 10.3724/SP.J.1146.2013.01955.
LIU Yang, JI Xinsheng, and LIU Caixia. Detection local community structure based on the identification of boundary nodes in complex networks[J]. *Journal of Electronics & Information Technology*, 2014, 36(12): 2809–2815. doi: 10.3724/SP.J.1146.2013.01955.

[12] SHAO Junming, HAN Zhichao, YANG Qinli, *et al*. Community detection based on distance dynamics[C]. Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, Sydney, 2015: 1075–1084.

[13] WANG Z and WANG Z. Research in social network based on rough set clustering algorithm[J]. *International Journal of Advancements in Computing Technology*, 2012, 4(15): 295–301.

[14] 朱文强, 伏玉琛. 一种基于粗糙集的社区结构发现算法[J]. 计算机工程, 2011, 37(14): 41–43. doi: 10.3969/j.issn.1000-3428.2011.14.012.
ZHU Wenqiang, and FU Shuchen. Community structure detection algorithm based on rough set[J]. *Computer Engineering*, 2011, 37(14): 41–43. doi: 10.3969/j.issn.1000-3428.2011.14.012.

[15] PAWLAK Z. Rough sets[J]. *International Journal of Computer & Information Sciences*, 1982, 11(5): 341–356.

[16] DüNTSCH I and GEDIGA G. Handbook of Cluster Analysis [M]. Chapman & Hall, 2016, Chapter 5.

[17] DONGEN S. A clustering algorithm for graphs[R]. CWI, Amsterdam, The Netherlands, 2000.

[18] RAND W M. Objective criteria for the evaluation of clustering methods[J]. *Journal of the American Statistical Association*, 1971, 66(336): 846–850.

[19] STREHL A, and GHOSH J. Cluster ensembles-a knowledge reuse framework for combining multiple partitions[J]. *Journal of Machine Learning Research*, 2002, 3(12): 583–617.

[20] ANDREA L, SANTO F, and FILIPPO R. Benchmark graphs for testing community detection algorithms[J]. *Physical Review E*, 2008, 78(4): 561–570.

[21] NEWMAN Mark. Network data[OL]. http://www-personal.umich.edu/~mejn/netdata/,2016.

张云雷：　男，1983 年生，博士，研究方向为数据挖掘、复杂网络、社交网络分析.

吴　斌：　男，1969 年生，教授，博士，研究方向为数据挖掘、复杂网络、社交网络分析、大数据分析.

刘　宇：　男，1986 年生，博士，研究方向为数据挖掘、复杂网络、社交网络分析.