基于平均互信息的最优社区发现方法

李东¹, 程鸣权¹, 徐杨^{1*}, 袁峰², 陈奕男¹, 付雅晴¹

1. 华南理工大学软件学院, 广州 510006

2. 广州中国科学院软件应用技术研究所, 广州 511458

* 通信作者. E-mail: xuyang@scut.edu.cn

收稿日期: 2018-06-08; 接受日期: 2018-09-06; 网络出版日期: 2019-05-15

国家自然科学基金 (批准号: 61602186) 和广东省科技计划项目 (批准号: 2015B010103002, 2016B050502001) 资助

摘要 本文提出一种基于平均互信息的最优社区发现方法 AMI (average mutual information), 该方法通过计算社区划分时的平均互信息值找出最优的社区划分. 将 AMI 方法作用在非重叠社区发现算法 GN 和重叠社区发现算法 COPRA 上分别获得改进的 AMI-GN 算法和 AMI-COPRA 算法. 将 AMI-GN 算法与 GN, FN, IE 算法进行对比实验, 实验结果表明 AMI-GN 算法相较于其他算法提高了社区发现的质量. 将 AMI-COPRA 算法与 COPRA, LPPB 算法进行对比实验, 实验结果表明 AMI-COPRA 算法大幅度提升原始 COPRA 算法的稳定性, 大大减少了平均迭代次数, 加快了算法的收敛速度. 相较于 LPPB 算法, 发现社区的质量相差不大, 但 AMI-COPRA 算法比 LPPB 算法更加稳定. 研究表明, 运用 AMI 方法可有效地改进典型的非重叠社区发现算法和重叠社区发现算法的性能.

关键词 AMI-COPRA 算法, AMI-GN 算法, 平均互信息, AMI 方法, 社区发现

1 引言

随着互联网、物联网技术的快速发展, 事物之间的联系更加紧密, 错综复杂的联系形成了多样、多变、规模庞大的网络, 即复杂网络^[1]. 近年来, 对复杂网络的研究已成为多个领域的研究热点之一^[2~4]. 对复杂网络进行社区发现既有助于发现整个网络的功能模块及其演化, 还有助于理解、分析复杂系统的拓扑结构及动力学特性^[5], 因此针对复杂网络的社区发现研究具有十分重要的理论意义和应用价值^[6,7]. 社区发现算法按照是否能够发现重叠社区可分为两类: 非重叠社区发现算法和重叠社区发现算法.

对于非重叠社区发现算法, 主要基于图划分思想和层次聚类思想. 其中, 基于图划分思想的典型算法有 KL (Kernighan-Lin) 算法^[8] 和谱二分法^[9], KL 算法主要使用贪婪策略, 通过优化社区外部的链

引用格式: 李东, 程鸣权, 徐杨, 等. 基于平均互信息的最优社区发现方法. 中国科学: 信息科学, 2019, 49: 613-629, doi: 10.1360/N112018-00149
Li D, Cheng M Q, Xu Y, et al. Optimal community detection method based on average mutual information (in Chinese). Sci Sin Inform, 2019, 49: 613-629, doi: 10.1360/N112018-00149

接和社区内部的链接从而优化初始社区划分结果, 谱二分法主要是基于 Laplace 特征向量. 但是, 这两种算法在进行社区发现时需要确定社区数目. 而基于层次聚类思想的社区发现算法, 比如 GN (Girvan-Newman) 算法^[10] 是基于最大边介数的分裂方式来构造可能的社区结构, 然后利用模块度来评判并选择最终的社区结构, 它在进行社区发现时并不需要确定社区数目, 这在一定程度上解决了 KL 算法和谱二分法的局限性. Newman 基于模块度提出了 FN (Fast-Newman) 算法^[11], 该算法在初始时每个节点单独作为一个社区, 然后对社区进行合并, 每次合并都是朝着 ΔQ 增量最大的方向进行, 所有的节点聚合成为一个大社区, 并且形成一个层次聚类树, 最后以模块度最大的方式对层次聚类树进行分割得到最终的社区结构. Deng 等^[12] 针对模块度应用在 GN 算法准确性不高这一缺陷, 从信息熵的角度提出了一种新的基于信息熵的模块度函数, 并且基于信息熵的模块度函数提出了 IE (information-entropy) 算法, IE 算法的基本思想是: 每去除一条介数最大的边, 就计算一次信息熵模块度值, 直到没有可去除的边为止, 然后在所有的模块度数值中寻找最大值, 这个最大值所对应的社区结构就是最终的社区结构. 层次聚类算法虽然划分结果相对准确且稳定, 但这类算法的时间复杂度都普遍较高, 并不适用于大规模的复杂网络.

对于重叠社区发现算法, 主要有传统的重叠社区发现算法和基于标签传播思想的重叠社区发现算法. 其中, 典型的传统重叠社区发现算法包括 CPM 算法^[13] 和 LFM 算法^[14]. CPM 算法假设社区由相互连接的完全子图 (clique) 组成, 每个节点可属于多个完全子图, 从而达到发现重叠社区的目的. LFM 算法通过随机选取初始节点, 然后设定合适的适应度函数, 根据适应度函数对初始节点进行扩充, 直到适应度函数为负时停止. 由于每个节点可能会被扩充到多个社区, 所以 LFM 算法能够发现重叠社区. 但是传统重叠社区发现算法时间复杂度较高, 实际的应用性不强. 而基于标签传播思想的重叠社区发现算法的时间复杂度可以趋近于线性, 这很好地解决了传统方法运算速度较慢的不足. 比如在简单快速的 LPA (label propagation algorithm) 算法^[15] 上改进的 COPRA (community overlap propagation algorithm) 算法^[16] 就能够快速地发现重叠社区. 该算法首先为每个节点初始化一组属于某个社区的标签对 (c, b) , c 表示社区标识, 初始化时每个节点的社区标识都是不一样的, b 表示隶属度, 任意节点的所有隶属度加起来都等于 1. 其次, 在标签更新的过程中, 设置一个阈值 $(1/v)$, v 表示每个节点最多可属于的社区个数. 在待更新节点的标签对中, 如果某个节点中存在隶属度大于等于阈值的标签对, 那么删除隶属度小于阈值的所有标签对, 并对保留下来的标签对进行归一化. 如果不存在隶属度大于等于阈值的标签对, 则选择隶属度最大的标签对作为该节点更新后的标签对, 如果有多个最大值则随机选择一个标签对作为更新后的标签对. 依次迭代直到满足终止条件. 此外, 因为 LPA 算法简单高效的特点, 一些学者^[17,18] 从不同的方面对标签传播算法进行了改进, 但都存在结果不稳定的问题^[19]. Wu 等^[20] 通过创建平衡系数使节点可拥有的标签数量不受限制, 提出了 BMLPA (balanced multi-label propagation algorithm) 算法, 从而提升 COPRA 算法的稳定性. Liu 等^[21] 在 COPRA 算法的基础上改进了网络的传播特性, 提出了 LPPB (label-propagation-probability-based) 算法, 该算法根据节点影响力确定更新顺序防止标签的“逆流”现象, 并引入节点的属性特征来影响标签传播的概率, 从而提升 COPRA 算法的准确性和稳定性. Li 等^[19] 针对重叠社区发现算法预先输入参数的局限性以及标签冗余等缺点, 提出了 MMCLPA (minimal maximal clique label propagation algorithm), 该算法通过寻找网络中的最小极大团 (MMC) 并对每个 MMC 中的节点赋予相同的标签来减少冗余标签, 提高算法的稳定性. 但是, MMCLPA 算法不是在 COPRA 算法的基础上进行改进的, 且其并没有针对标签传播阶段进行改进. Zhang 等^[22] 虽然在传统 COPRA 算法中引入信息熵和局部相关性提出了 COPRA-EP 算法, 但是 COPRA-EP 算法是利用信息熵为节点产生更新序列指导标签更新. 而本文与它不同的是将信息熵作用于标签选择阶段.

此外,有一些学者根据社区网络中节点语义信息内容(如微博、社会标签等),将具有相似信息内容的节点划分为同一社区^[23].主要的算法有:Nagarajan 等^[24]提出的 LCM (link-content model) 模型,该模型将用户间的共享信息和传递信息作为 link-content,将 link 作为社区分类的基本单元来构建 LDA 模型^[25].文献[26]利用用户标签的使用频率等用户兴趣信息,来达到将兴趣相似的用户加入到同一个社区的目的.文献[27~31]利用 LDA 模型对语义网络进行主题挖掘建立语义空间,分别利用标签传播算法、BLOCK 场、随机游走以及话题综合因子分解等方法进行社区发现.还有一些学者用低维、稠密、实值的向量表示网络中的节点,进而将异质信息投影到同一个低维空间中,方便进行随机游走等计算^[32,33].

本文的贡献主要有以下 3 点:(1)提出了一种基于平均互信息的最优社区发现方法 AMI;(2)基于 AMI 方法分别提出了非重叠社区发现算法 GN 和重叠社区发现算法 COPRA 的改进算法 AMI-GN 和 AMI-COPRA;(3)通过将 AMI-GN 算法与 GN, FN, IE 算法进行对比实验,证明了 AMI-GN 算法有较高的准确性.通过将 AMI-COPRA 算法与 COPRA 和 LPPB 算法进行对比实验,证明 AMI 方法作用于重叠社区发现算法时,能够增强算法的稳定性,加快算法的收敛速度.实验证明了运用 AMI 方法可有效地改进典型的非重叠社区发现算法和重叠社区发现算法的性能.

2 AMI 社区发现方法

本文提出了一种基于平均互信息 (average mutual information, 简称 AMI) 的最优社区发现方法. AMI 方法首先通过计算每次社区划分的平均互信息值 (I_P). 然后,选取 I_P 值最大的社区划分作为最优的社区划分,在此基础上可对现有的社区发现算法进行改进. I_P 的计算公式如下:

$$I_P = \omega \cdot I(X;Y), \quad (1)$$

其中 X 表示社区划分前的社区结构, Y 表示社区划分后的社区结构, ω 表示社区结构 X 和 Y 的关联性, $I(X;Y)$ 表示社区结构 X 和 Y 的平均互信息.

社区发现算法按照是否能够发现重叠社区可分为两类:非重叠社区发现算法和重叠社区发现算法.为了验证 AMI 方法可以作用于不同种类的社区发现算法,本文将 AMI 方法分别作用在非重叠社区发现算法和重叠社区发现算法上,通过实验证明 AMI 方法可有效地改进典型的非重叠社区发现算法和重叠社区发现算法的性能,进而证明 AMI 方法的有效性.

由于不同种类的社区发现算法有不同的特性,所以将 AMI 方法作用于具体的社区发现算法时,依据算法拥有的特性不同,平均互信息值 I_P 也会有所不同. $I(X;Y)$ 按照社区划分后的社区结构 Y 是否可以由划分前的社区结构 X 中某个社区分裂(或某两个社区凝聚)得到,将 $I(X;Y)$ 的计算公式分为两种:

$$I(X;Y) = \begin{cases} \sum_i \sum_j I(X_i;Y_j), \\ H(X) - H(X|Y). \end{cases}$$

当社区划分后的社区结构 Y 可由划分前的社区结构 X 中某个社区分裂(或某两个社区凝聚)得到,在计算平均互信息 $I(X;Y)$ 时,使用 $I(X;Y) = \sum_i \sum_j I(X_i;Y_j)$;当社区划分后的社区结构 Y 不可由划分前的社区结构 X 中某个社区分裂(或某两个社区凝聚)得到,在计算平均互信息 $I(X;Y)$ 时,则使用 $I(X;Y) = H(X) - H(X|Y)$.

图 1 社区划分的两种情况. (a) 社区没有发生分裂; (b) 社区分裂成 2 个社区

Figure 1 Two cases of the community partition. (a) No community splits; (b) community splits into two communities

2.1 基于层次聚类的非重叠社区发现

由于基于层次聚类的非重叠社区发现算法在进行社区划分时, 社区划分后的社区结构 Y 可以由划分前的社区结构 X 中某个社区分裂 (或某两个社区凝聚) 得到的. 所以, 将 AMI 方法作用在基于层次聚类的非重叠社区发现算法时, 社区划分的平均互信息值 (I_P) 的计算公式变换为

$$I_P = \sum_i \sum_j \omega_{ij} \cdot I(X_i; Y_j), \quad (2)$$

其中 X_i 表示划分前的社区结构中第 i ($i = 1, 2, 3, \dots, n$) 个社区, Y_j 表示划分后的社区结构中第 j ($j = 1, 2, 3, \dots, m$) 个社区. $I(X_i; Y_j)$ 表示社区 X_i 与 Y_j 的平均互信息值, $\omega_{ij} = P(X_i, Y_j)$ 表示社区 X_i 和 Y_j 的关联性, $P(X_i, Y_j) = P(Y_j|X_i) \times P(X_i)$, $P(Y_j|X_i)$ 表示在社区 X_i 的点被划分到社区 Y_j 的概率, $P(X_i)$ 表示网络中的点被划分到社区 X_i 的概率.

下面以 AMI 方法作用于 GN 算法为例, 说明 AMI 方法如何作用在基于层次聚类的非重叠社区发现算法上. AMI 方法具体是对 GN 算法选择社区结构这一过程进行改进, 利用平均互信息 (I_P) 最大值找出优化的社区划分, 进而找出最优的社区结构. 由于 GN 算法是基于最大边介数的分裂方式来构造多种社区结构的. 于是, 对于每次划分前、后的社区 X_i 和 Y_j 的关联性 ω_{ij} , 只需考虑以下两种情况 (如图 1):

- (1) 在社区划分中, 某社区没有发生分裂;
- (2) 在社区划分中, 某社区分裂成其他两个社区.

针对上述两种情况, 分别计算 $P(Y_j|X_i)$.

对于第 1 种情况:

$$P(Y_j|X_i) = \begin{cases} 1, & \text{当 } Y_j \text{ 和 } X_i \text{ 为同一社区,} \\ 0, & \text{当 } Y_j \text{ 和 } X_i \text{ 为不同社区.} \end{cases}$$

对于第 2 种情况:

$$P(Y_j|X_i) = \begin{cases} \frac{n_{y_i}}{n_{x_i}}, & \text{当 } Y_j \text{ 是社区 } X_i \text{ 分裂得到的,} \\ 0, & \text{当 } Y_j \text{ 不是社区 } X_i \text{ 分裂得到的,} \end{cases}$$

其中 n_{x_i} 表示社区 X_i 中的节点总数, n_{y_j} 表示社区 Y_j 中的节点总数.

因为 $P(X_i)$ 表示网络中的点被划分到社区 X_i 的概率, 所以 $P(X_i)$ 的计算公式如下:

$$P(X_i) = \frac{n_{x_i}}{n},$$

其中 n 表示网络中的节点总数.

利用 $P(X_i, Y_j) = P(Y_j|X_i) \times P(X_i)$, 将式 (2) 展开为

$$I_P = P(X_1, Y_1)I(X_1, Y_1) + \cdots + P(X_i, Y_i)I(X_i, Y_i) + P(X_i, Y_{i+1})I(X_i, Y_{i+1}),$$

其中上述公式表明社区 X_i 在本次划分中被分裂成了 Y_i 和 Y_{i+1} 这两个社区. 接下来, 对上述公式中的 $I(X_i; Y_j)$ 进行分析, $P(X_i = 0)$ 表示节点不属于社区 X_i 的概率, $P(X_i = 1)$ 表示节点属于社区 X_i 的概率, 其计算公式如下:

$$P(X_i = 0) = \frac{n - n_{x_i}}{n}, \quad (3)$$

$$P(X_i = 1) = \frac{n_{x_i}}{n}. \quad (4)$$

同理 $P(Y_j = 0)$ 和 $P(Y_j = 1)$ 的计算公式如下:

$$P(Y_j = 0) = \frac{n - n_{y_j}}{n}, \quad P(Y_j = 1) = \frac{n_{y_j}}{n}.$$

$P(Y_j = 1|X_i = 1)$ 表示在节点属于社区 X_i 的条件下, 该节点也属于社区 Y_j 的概率, 则 $P(Y_j = 1|X_i = 1)$ 的计算公式如下:

$$P(Y_j = 1|X_i = 1) = \frac{n_{y_j}}{n_{x_i}}.$$

依次类推 $P(Y_j = 0|X_i = 1)$ 的计算公式如下:

$$P(Y_j = 0|X_i = 1) = \frac{n_{x_i} - n_{y_j}}{n_{x_i}}.$$

因为社区 Y_j 和 X_i 要么是同一个社区, 要么社区 Y_j 就是由社区 X_i 分裂得到的, 所以某节点如果不在社区 X_i 中, 那么必然不会在社区 Y_j 中. 所以, $P(Y_j = 0|X_i = 0) = 1$, $P(Y_j = 1|X_i = 0) = 0$.

根据平均互信息的计算公式, 则 $I(X_i; Y_j)$ 的计算表达式如下:

$$I(X_i; Y_j) = \sum_a \sum_b P(X_i = a, Y_j = b) [\log_2 P(X_i = a, Y_j = b) - (\log_2 P(X_i = a) + \log_2 P(Y_j = b))],$$

其中 a 和 b 分别可以取 0 或 1. 并且, 利用上述定义, 根据公式 $P(X_i = x, Y_j = y) = P(Y_j = y|X_i = x) \times P(X_i = x)$, 求得 $P(X_i = x, Y_j = y)$ 的值, 从而计算出 $I(X_i; Y_j)$, 进而计算出 I_P 的值.

为了找到最终的社区结构, 要对 I_P 值最大的社区划分前、后的社区结构分别计算信息熵. 社区结构的信息熵计算公式如下所示:

$$H(Z) = - \sum_{z_i} \log_2 P(z_i = 0) - \sum_{z_i} \log_2 P(z_i = 1),$$

其中随机变量 Z 表示一种社区结构, z_i 表示社区结构 Z 中的第 i 个社区, $P(z_i = 0)$ 表示节点不属于社区 z_i 的概率 (见式 (3)), $P(z_i = 1)$ 表示节点属于社区 z_i 的概率 (见式 (4)).

最后, 选择信息熵值较小的社区结构, 即为最终的社区结构.

AMI-GN 算法的具体步骤如下所示, 本算法的伪代码如算法 1 所示.

Step 1. 计算网络中的边介数, 并删除介数最大的边.

Step 2. 重复 Step 1 直到网络中的社区发生分裂.

Step 3. 计算本次社区划分的平均互信息值 (I_P).

Step 4. 判断网络中是否还有可分裂的社区. 如果有, 则进入 Step 1. 如果没有, 则进入 Step 5.

Step 5. 找出平均互信息的最大值 (Max_I_P), 并记录 Max_I_P 对应社区划分前、后的社区结构.

Step 6. 分别计算社区划分前、后社区结构的信息熵, 信息熵值较小的社区结构即为最终的社区结构.

Algorithm 1 AMI-GN

Input: The network represented by the form $G(E, V)$, and E represents the total number of nodes in the network, V represents the total number of edges in the network;

Output: Optimal community structure Max_CS in the network;

```

1:  $\text{Max}_I_P = \text{INT\_MIN}$ ;
2: for each  $i$  in  $E$  do
3:   Calculate all edge betweenness and save in array  $\text{EdgeBetweenness}$ ;
4:   while No community splits do
5:     Delete edge with the highest  $\text{EdgeBetweenness}$  value;
6:   end while
7:   Calculate  $I_P$ ;
8:   if  $I_P > \text{Max}_I_P$  then
9:      $\text{Max}_I_P = I_P$ ;
10:    Record the before and after community structure of community partition corresponding to  $\text{Max}_I_P$ ;
11:   end if
12: end for
13: Calculate the information entropy of the before and after community structure of community partition corresponding to  $\text{Max}_I_P$  and save in  $\text{Entropy\_Before}$  and  $\text{Entropy\_After}$  separately;
14: if  $\text{Entropy\_Before} > \text{Entropy\_After}$  then
15:   Save the community structure of after this partition into  $\text{Max\_CS}$ ;
16: else
17:   Save the community structure of before this partition into  $\text{Max\_CS}$ ;
18: end if
19: return  $\text{Max\_CS}$ ;

```

2.2 基于标签传播的重叠社区发现

由于基于标签传播的重叠社区发现算法在进行社区划分时, 社区划分后的社区结构 Y 不能由划分前的社区结构 X 中某个社区分裂 (或某两个社区凝聚) 得到. 所以, 将 AMI 方法作用在基于标签传播的重叠社区发现算法时, 社区划分的平均互信息值 (I_P) 的计算公式变换为

$$I_P = \omega \cdot [H(X) - H(X|Y)],$$

其中 X 表示社区划分前的社区结构, Y 表示社区划分后的社区结构, $H(X)$ 表示社区结构 X 的信息熵, $H(X|Y)$ 表示在社区结构 Y 选定的条件下社区结构 X 的信息熵.

下面以 AMI 方法作用于 COPRA 算法为例, 说明 AMI 方法如何作用在基于标签传播的重叠社区发现算法上. 由于 COPRA 算法在标签选择过程中存在很大的随机性, 即每个点在社区划分时所属的社区也是随机的. 所以, 对于某个复杂网络的每一次社区划分过程中, 任意两个社区 X_i 和 Y_j 之间的关联性也是随机的. 所以, 我们可以假设: 对于某个复杂网络的每一次社区划分过程, 任意两个社区 X_i 和 Y_j 之间的关联性为某个常数, 即 $\omega = L$. 于是, COPRA 算法下的平均互信息值 I_P 的计算公式如下:

$$I_P = L \cdot [H(X) - H(X|Y)], \quad (5)$$

图2 标签初始化

Figure 2 Label initialization

图3 标签更新阶段

Figure 3 Label update phase

其中 L 表示某个常数.

AMI 方法具体对 COPRA 算法的标签传播这一阶段进行改进. 对于 COPRA 算法中需要随机选择标签的节点, 我们选择平均互信息值 (I_P) 最大的标签作为节点的标签. 如果存在多个平均互信息 (I_P) 最大值, 再随机选取一个. 在标签传播过程中, $H(X)$ 表示标签传播前的社区结构 X 的信息熵, 因为标签传播前的社区结构 X 是不变, 即 $H(X)$ 是一个固定值, 由式 (5) 可知: 在 $H(X)$ 不变的情况下, 要想使得 I_P 最大, 就是要使得 $H(X|Y)$ 最小. 于是, 依据 $H(X|Y)$ 最小来对每个节点进行标签选择.

下面举一个例子详细说明 AMI 方法如何改进标签传播阶段. 首先, 对标签进行初始化 (如图 2), 为每个节点赋予一个标签对 (c, b) , c 表示该节点所属的社区标识, b 表示节点属于该标签的隶属度. 在标签更新阶段 (如图 3), 每个节点的候选标签都来自于它的邻居节点.

接下来, 对标签进行更新, 即确定每个点在本次更新后的标签对. 如果某个节点中最大隶属度所对应的标签对有多且都小于阈值 $1/v$. 先将确定最终标签对的节点更新完成后, 再利用 AMI 方法依次对待更新节点中的标签对进行选择. 对于图 3 中的情况, 假设每个节点最多可属于社区的个数 (v) 为 2. 那么, 在进行标签更新的时候, 由于节点 3 的标签对 $(b, 1/2)$, $(d, 1/2)$ 和节点 6 的标签对 $(e, 1/2)$, $(g, 1/2)$ 的隶属度都大于等于 $1/2$, 所以节点 3 和 6 的标签对都是更新完的状态. 由于节点 1, 2, 4, 5 和 7 的标签对隶属度都小于 $1/2$, 所以节点 1, 2, 4, 5 和 7 的标签对都是处于待更新的状态. 从节点 1 开始依次对节点的标签进行更新. 对于节点 1, 它的待更新标签对有 4 个, 分别是 $(b, 1/4)$, $(d, 1/4)$, $(e, 1/4)$ 和 $(g, 1/4)$, 依次计算节点 1 选择 $(b, 1/4)$, $(d, 1/4)$, $(e, 1/4)$, $(g, 1/4)$ 标签对后的平均互信息 (I_P) 值, 而 I_P 值的计算公式为 $I_P = l \cdot [H(X) - H(X|Y)]$, $H(X)$ 表示标签传播前的社区结构 X 的信息熵. 在本例中社区结构 X 就是标签初始化的社区结构. 对于节点 1 在标签更新过程中选择 $(b, 1/4)$, $(d, 1/4)$, $(e, 1/4)$, $(g, 1/4)$ 任一标签对, $H(X)$ 的值都是一个定值, 要使得 I_P 值最大, 只需要 $H(X|Y)$ 的值最小即可. 于是, 计算节点 1 选择 $(b, 1/4)$, $(d, 1/4)$, $(e, 1/4)$, $(g, 1/4)$ 任一标签对的 $H(X|Y)$ 的值, 节点 1 选择各个标签对的 $H(X|Y)$ 都为 1.7925, 因为 $H(X|Y)$ 的最小值有多个, 则随机选择一个. 如果 $H(X|Y)$ 的最小值只有一个, 那么选择该值即可. 最后, 在每次标签传播的更新标签阶段都使用 AMI 方法, 直到标签传播算法达到迭代终止条件为止.

AMI-COPRA 算法的具体步骤如下所示, 其中基于信息熵的标签更新算法如算法 2 所示.

Step 1. 初始化节点的标签对.

Step 2. 在每次标签传播过程中, 判断每个节点中是否存在隶属度大于等于阈值 ($\psi = 1/v$, v 表示节点可同时属于的最大社区个数) 的标签对.

- Step 3. 如果存在, 删除所有隶属度小于阈值的标签对, 对保留下来的标签对进行归一化.
- Step 4. 如果不存在, 通过基于信息熵的标签更新算法为该节点更新标签对.
- Step 5. 重复 Steps 2~4, 直到满足迭代终止条件.
- Step 6. 拥有相同社区标识的节点组成一个社区, 从而得到最终的社区结构.

Algorithm 2 Label update algorithm based on information entropy

Input: Updated node set (updatedNodeMap) and label set to be updated (labelMap);

Output: Updated label pair (updateLabelPair);

```
1: minEntropy=Float.MAX_VALUE;
2: for each label in labelMap do
3:   label_c=label.getLabelC();
4:   if updatedNodeMap.containsKey(label_c) then
5:     updatedNodeMap.put(label_c, Float.valueOf(1)+(float)updatedNodeMap.get(key));
6:   else
7:     updatedNodeMap.put(label_c, Float.valueOf(1));
8:   end if
9:   for each label_c in updatedNodeMap do
10:    currentNodeNumber = updatedNodeMap.get(label_c);
11:     $x = \text{currentNodeNumber} / \text{totalNodeNumber}$ ;
12:    currentEntropy+ = (float)( $x \cdot (-1) \cdot \text{Math.log}(x) / \text{Math.log}(2)$ );
13:   end for
14:   if currentEntropy < minEntropy then
15:     minEntropy = currentEntropy;
16:     Record corresponding tag pair and Assign to updateLabelPair;
17:   end if
18: end for
19: return updateLabelPair.
```

3 实验分析

3.1 AMI-GN 算法实验

为验证 AMI-GN 算法的性能, 我们将 AMI-GN 与其他基于层次聚类的非重叠社区发现算法 GN, FN, IE 进行比较. AMI-GN 算法的实验测试分为两部分: (1) 将 AMI-GN 算法与原始 GN 算法、基于模块度的 FN 算法、基于信息熵模块度函数的 IE 算法进行划分准确度对比实验. (2) 将 AMI-GN 算法划分结果与目前认可的划分结果进行比较分析.

3.1.1 实验数据与评价指标

因为基于层次聚类的非重叠社区发现算法的时间复杂度普遍偏高, 并不适用于大规模的复杂网络. 所以在对 AMI-GN 算法的实验中采用的数据集规模相对较小, 大规模的数据集将用于后面对 AMI-COPRA 算法的实验中. 表 1 给出了实验中所用数据集的概要信息.

因为上述的实验数据集都是有目前认可的划分结果的, 所以采用目前认可的社区划分个数 (CN) 和标准化互信息 (NMI, normalized mutual information) 这两个重要参数作为评判指标来衡量社区发现结果的质量.

表 1 实验数据集
Table 1 Experiment dataset

Dataset	Node number	Edge number	Community number
Karate	34	78	2
Dolphin	62	159	2
Polbooks	105	441	3

表 2 AMI-GN 与其他 3 种经典算法比较结果
Table 2 Comparison of AMI-GN with the other three classical algorithms

Dataset	Index	AMI-GN	GN	FN	IE
Karate	CN	2	5	2	2
	NMI	0.84	0.58	0.84	0.84
Dolphin	CN	2	5	3	2
	NMI	0.89	0.55	0.65	0.89
Polbooks	CN	3	5	3	2
	NMI	0.58	0.56	0.57	0.55

标准化互信息指标 NMI 用来衡量算法得到的社区划分结果与实际社区划分结果之间的相似度, 其计算公式如下:

$$NMI = \frac{-2 \sum_{i=1}^{C_A} \sum_{j=1}^{C_B} C_{ij} \log_2(C_{ij}N/C_{i.}C_{.j})}{\sum_{i=1}^{C_A} C_{i.} \log_2(C_{i.}/N) + \sum_{j=1}^{C_B} C_{.j} \log_2(C_{.j}/N)},$$

其中 A 和 B 分别表示网络的两种划分, C 是混淆矩阵 (confusion matrix), 混淆矩阵 C 中的元素 C_{ij} 表示既在 A 划分的社区 i 中同时也在 B 划分的社区 j 中的节点数目. C_A (C_B) 表示 A (B) 划分中的社区个数, $C_{i.}$ ($C_{.j}$) 是混淆矩阵 C 中第 i 行 (j 列) 元素之和, N 是原始网络中节点的总数目.

3.1.2 实验结果

表 2 给出了 AMI-GN, GN, FN 与 IE 算法在划分正确度上的实验结果, 其中 FN 算法在 Karate, Dolphin 和 Polbooks 数据集上计算的 NMI 值来源于文献 [34]. 分析表 2 可知: AMI-GN 算法在 Karate, Dolphin, Polbooks 数据集上划分所得的社区个数 (CN) 和目前认可的社区个数一致, 这一点优于 GN, FN 和 IE 算法. 进一步, 对于这 3 个数据集的 NMI 值来说, AMI-GN 算法所得到的 NMI 值大于其他算法的 NMI 值, 这说明 AMI-GN 算法得到的社区划分结果比其他算法的准确度更高.

虽然 AMI-GN 算法的 NMI 值要高于其他 3 种算法的 NMI 值, 但是 NMI 值还没有达到 1. 可进一步对 AMI-GN 算法划分结果和目前认可的划分结果进行比较分析. 表 3 给出了 AMI-GN 算法的社区划分与目前认可的社区划分的比较结果, 由表 3 知: 在 Karate 和 Dolphin 这 2 个数据集中, 不同的节点各是 1 个, 分别是 Karate 数据集中的节点 3 和 Dolphin 数据集中的节点 40. 而对于 Polbooks 数据集, AMI-GN 算法的社区划分与目前认可的社区划分结果有 17 个节点的不同.

对 Karate 和 Dolphin 数据集中不同节点的链接进行分析. 图 4 是 Karate 数据集和 Dolphin 数据集的不同节点划分示意图. 由图 4(a) 和 (b) 看出, Karate 数据集中的节点 3 和 Dolphin 数据集中的节点 40 连接两个社区的链接数都相等, 又因为该网络是无权重的网络, 说明节点 3 和 40 是既可以属

表 3 AMI-GN 算法的划分结果与目前认可的划分结果的比较结果
Table 3 Comparison of AMI-GN algorithm with currently approved partition result

Dataset	Total number of different nodes	Different node index
Karate	1	3
Dolphin	1	40
Polbooks	17	1, 5, 7, 8, 19, 29, 47, 49, 53, 59, 65, 66, 69, 77, 78, 104, 105

图 4 (网络版彩图) 不同节点的划分示意图. (a) 在 Karate 数据集下; (b) 在 Dolphin 数据集下
Figure 4 (Color online) Schematic of different nodes. (a) Karate dataset; (b) Dolphin dataset

于社区 1 也可以属于社区 2 的, 这说明 AMI-GN 算法对 Karate 和 Dolphin 数据集的划分结果也是正确的.

对 Polbooks 数据集中不同节点的链接进行分析, 如表 4 所示. 其中 Links in community (R) 表示公认划分结果中社区内的链接数, Links between community (R) 表示公认划分结果中社区间的链接数, Links in community (A) 表示 AMI-GN 算法的划分结果中社区内的链接数, Links between community (A) 表示 AMI-GN 算法的划分结果中社区间的链接数.

对于节点 1, 在目前认可的划分结果中, 节点 1 连接其他节点的总链接数 (简称节点 1 的总链接数) 为 6, 节点 1 与它所属社区内部节点相连的链接数 (简称节点 1 的社区内链接数) 为 2, 而节点 1 连接其他社区中节点的链接数 (简称节点 1 的社区间链接数) 分别为 4 和 0. 在 AMI-GN 算法的划分结果中, 节点 1 的社区内链接数为 6, 社区间链接数分别为 0 和 0, 根据社区结构具有的同社区内节点相互连接紧密、异社区间节点相互连接稀疏的特性^[10], 从而说明节点 1 在 AMI-GN 算法中的划分结果是优于目前认可的划分结果. 同理可以说明节点 5, 7, 8, 19, 29, 47, 49, 59, 65, 66, 77, 78, 104 在 AMI-GN 算法中的划分结果是优于目前认可的划分结果.

对于节点 53, 在目前认可的划分结果中, 节点 53 的总链接数为 5, 社区内链接数为 3, 社区间链接数为 1 和 1, 而在 AMI-GN 算法的划分结果中, 节点 53 的社区内链接数也为 3, 社区间链接数分别为 2 和 0, 这说明节点 53 在 AMI-GN 算法中的划分结果不差于目前认可的划分结果. 同理可以说明节点 69, 105 在 AMI-GN 算法中的划分结果不差于目前认可的划分结果. 进而说明 AMI-GN 算法对 Polbooks 数据集的划分结果要优于目前认可的划分结果.

3.2 AMI-COPRA 算法实验

为验证 AMI-COPRA 算法的性能, 我们将 AMI-COPRA 与其他基于标签传播的重叠社区发现算法 COPRA, LPPB 进行对比实验.

表 4 Polbooks 数据集中不同节点的链接分析
Table 4 Link analysis of different node in Polbooks dataset

Node	Links in community (R)	Links between community (R)	Links in community (A)	Links between community (A)
1	2	4, 0	6	0, 0
5	3	3, 2	5	3, 0
7	4	7, 0	11	0, 0
8	1	4, 3	4	3, 1
19	1	2, 0	3	0, 0
29	1	2, 0	2	1, 0
47	0	3, 1	3	1, 0
49	0	4, 0	4	0, 0
53	3	1, 1	3	2, 0
59	5	5, 3	6	4, 3
65	5	2, 2	6	3, 0
66	4	2, 1	5	2, 0
69	3	1, 0	3	1, 0
77	0	3, 10	11	2, 0
78	2	5, 0	5	1, 1
104	1	1, 0	2	0, 0
105	2	1, 0	2	1, 0

表 5 实验数据集
Table 5 Experiment dataset

Dataset	Reference	Node number	Edge number
Karate	[34]	34	78
Dolphin	[34]	62	159
Football	[34]	115	613
Netscience	[35]	379	914
Protein-protein	[13]	2445	6265
Blogs	[36]	3982	6803
PGP	[37]	10680	24316
Blogs2	[36]	30557	82301
Cond-mat-2003	[16]	27519	116181

3.2.1 实验数据与评价指标

实验采用的真实网络数据集如表 5 [13, 16, 34~37] 所示. 因为上述大部分实验数据集都没有给定社区划分结果, 所以本实验采用评价重叠社区结构的模块度 (Q_{ov} [38]) 平均值、模块度 (Q_{ov}) 标准差以及平均迭代次数作为评判指标分析 AMI-COPRA 算法的性能. 评价重叠社区结构的模块度 Q_{ov} 的计算公式如下:

$$Q_{ov} = \frac{1}{2m} \sum_{c \in C} \sum_{i,j \in V} \left[q_{ijc} A_{ij} - p_{ijc} \frac{k_i k_j}{2m} \right],$$

表 6 AMI-COPRA 与其他两种算法比较结果
Table 6 Comparison of AMI-COPRA with other two algorithms

Dataset	Algorithm	Average Q_{ov}	Standard deviation Q_{ov}	Average number of iterations
Karate	COPRA	0.459	0.244	11.3
	LPPB	0.733	0	—
	AMI-COPRA	0.720	0	7
Dolphin	COPRA	0.665	0.069	13.1
	LPPB	0.702	0.018	—
	AMI-COPRA	0.728	0	6
Football	COPRA	0.691	0.030	7.6
	LPPB	—	—	—
	AMI-COPRA	0.707	0.020	5.8
Netscience	COPRA	0.812	0.026	24.3
	LPPB	—	—	—
	AMI-COPRA	0.841	0.006	15
Protein-protein	COPRA	0.552	0.066	56.2
	LPPB	—	—	—
	AMI-COPRA	0.710	0.001	73.4
Blogs	COPRA	0.748	0.007	166.8
	LPPB	0.756	0.002	—
	AMI-COPRA	0.765	0	151
PGP	COPRA	0.788	0.017	249.6
	LPPB	0.796	0.009	—
	AMI-COPRA	0.793	0	213
Blogs2	COPRA	0.604	0.032	172.5
	LPPB	—	—	—
	AMI-COPRA	0.608	0.005	53
Cond-mat-2003	COPRA	0.675	0.057	331.4
	LPPB	—	—	—
	AMI-COPRA	0.687	0.005	61.7

其中 q_{ijc} 表示节点 i 和 j 都属于社区 c 的概率, p_{ijc} 表示节点 i 属于社区 c 或者节点 j 属于社区 c 的概率.

3.2.2 实验结果

将 3 种算法应用在 9 种真实网络数据集上进行社区发现, 并在模块度 (Q_{ov}) 平均值、模块度 (Q_{ov}) 标准差以及平均迭代次数方面进行对比. 实验结果是对每个真实网络做 100 次实验的平均结果, 如表 6 所示. 其中, 表 6 中的 “—” 表示原论文未给出实验结果, 加粗部分表示 3 种算法中最优的实验结果. COPRA 算法在 Netscience, Protein-protein, Blogs, PGP, Blogs2 和 Cond-mat-2003 数据集上的实验数据来源于文献 [16], LPPB 算法在 Karate, Dolphin, Blogs 和 PGP 数据集上的实验数据来源于文献 [21].

为了使得实验结果更清晰明了, 于是分别根据 3 种评判指标对不同算法的实验结果作图分析. 图 5 表示 AMI-COPRA 与其他两种算法在模块度 (Q_{ov}) 平均值上的比较结果, 图 6(a) 表示 AMI-

图 5 (网络版彩图) AMI-COPRA 与其他两种算法在模块度 Q_{ov} 平均值上的比较结果
 Figure 5 (Color online) Comparison of AMI-COPRA with other two algorithms on average Q_{ov} value

图 6 (网络版彩图) AMI-COPRA 与 COPRA 算法的比较结果. (a) 在模块度 (Q_{ov}) 标准差上; (b) 在平均迭代次数上

Figure 6 (Color online) Comparison result of AMI-COPRA with COPRA algorithm. (a) Standard deviation Q_{ov} ; (b) average number of iterations

COPRA 与 COPRA 算法在模块度 (Q_{ov}) 标准差上的比较结果, 图 6(b) 表示 AMI-COPRA 与 COPRA 算法在平均迭代次数上的比较结果. 对实验结果进行分析可知:

(1) 通过分析 AMI-COPRA 算法的模块度 (Q_{ov}) 平均值, 发现 AMI-COPRA 算法比原始 COPRA 算法在模块度 (Q_{ov}) 平均值上提升了 0.004~0.261 不等, 证明 AMI-COPRA 算法提高了原始 COPRA 算法社区发现的质量;

(2) 通过分析 AMI-COPRA 算法的模块度 (Q_{ov}) 标准差, 发现 AMI-COPRA 算法的模块度 (Q_{ov}) 标准差非常小, 进而证明 AMI-COPRA 算法大幅度提高了原始 COPRA 算法的稳定性, 很好地解决了原始 COPRA 算法不稳定的缺陷;

(3) 通过分析 AMI-COPRA 算法的平均迭代次数, 发现 AMI-COPRA 算法降低了原始 COPRA 算法的平均迭代次数, 说明 AMI-COPRA 算法加快了原始算法的收敛速度;

(4) 将 AMI-COPRA 算法与 LPPB 算法进行对比实验后, 虽然在模块度 (Q_{ov}) 平均值上与 LPPB 算法相差不大, 但 AMI-COPRA 算法比 LPPB 算法更加稳定. 进而说明 AMI-COPRA 算法更适用于实际的复杂网络.

综上所述, 运用 AMI 方法可有效地改进典型的重叠社区发现算法的性能.

4 结束语

本文提出了一种基于平均互信息的最优社区发现方法, 并将 AMI 方法作用在基于层次聚类的非重叠社区发现算法 GN 和基于标签传播的重叠社区发现算法 COPRA 上, 分别获得改进的 AMI-GN 算法和 AMI-COPRA 算法. 将 AMI-GN 算法与原始 GN, FN, IE 算法在 3 种真实网络数据集上进行了比较, 实验结果表明 AMI-GN 算法在划分个数和 NMI 值上都明显优于 GN, FN 和 IE 算法. 将 AMI-COPRA 算法与原始 COPRA, LPPB 在 9 种真实网络数据集上进行比较分析, 实验结果表明 AMI-COPRA 算法提高了社区发现的质量, 大幅度增强了原始算法的稳定性, 大大加快了算法收敛的速度. 因此, 运用 AMI 方法可有效地改进典型的非重叠社区发现算法和重叠社区发现算法的性能. 下一步将着重研究如何将 AMI 方法作用于动态网络社区发现.

参考文献

- 1 Qiao S J, Han N, Zhang K F, et al. Algorithm for detecting overlapping communities from complex network big data. *J Softw*, 2017, 28: 631–647 [乔少杰, 韩楠, 张凯峰, 等. 复杂网络大数据中重叠社区检测算法. *软件学报*, 2017, 28: 631–647]
- 2 Bai L, Cheng X Q, Liang J Y, et al. Fast graph clustering with a new description model for community detection. *Inf Sci*, 2017, 388–389: 37–47
- 3 Atay Y, Koc I, Babaoglu I, et al. Community detection from biological and social networks: a comparative analysis of metaheuristic algorithms. *Appl Soft Comput*, 2017, 50: 194–211
- 4 Liu B Y, Wang C R, Wang C, et al. Microblog community discovery algorithm based on dynamic topic model with multidimensional data fusion. *J Softw*, 2017, 28: 246–261 [刘冰玉, 王翠荣, 王聪, 等. 基于动态主题模型融合多维数据的微博社区发现算法. *软件学报*, 2017, 28: 246–261]
- 5 Yang G, Zheng W P, Wang W J, et al. Community detection algorithm based on weighted dense subgraphs. *J Softw*, 2017, 28: 3103–3114 [杨贵, 郑文萍, 王文剑, 等. 一种加权稠密子图社区发现算法. *软件学报*, 2017, 28: 3103–3114]
- 6 Huang F L, Yu G, Zhang J L, et al. Mining topic sentiment in micro-blogging based on micro-blogger social relation. *J Softw*, 2017, 28: 694–707 [黄发良, 于戈, 张继连, 等. 基于社交关系的微博主题情感挖掘. *软件学报*, 2017, 28: 694–707]
- 7 Wilson S J, Wilkins A D, Lin C H, et al. Discovery of functional and disease pathway by community detection protein-protein interaction networks. *Pac Symp Biocomput*, 2016, 22: 336–347

- 8 Kernighan B W, Lin S. An efficient heuristic procedure for partitioning graphs. *Bell Syst Technical J*, 1970, 49: 291–307
- 9 Barnes E R. An algorithm for partitioning the nodes of a graph. *Siam J Alg Disc Math*, 1982, 3: 303–304
- 10 Girvan M, Newman M E J. Community structure in social and biological networks. In: *Proceedings of the National Academy of Sciences of the United States of America*, 2002. 99: 7821–7826
- 11 Newman M E J. Fast algorithm for detecting community structure in networks. *Phys Rev E*, 2004, 69: 066133
- 12 Deng X, Wang B, Wu B, et al. Research and evaluation on modularity modeling in community detecting of complex network based on information entropy. In: *Proceedings of the 3rd IEEE International Conference on Secure Software Integration and Reliability Improvement*, 2009. 297–302
- 13 Palla G, Derényi I, Farkas I, et al. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 2005, 435: 814–818
- 14 Lancichinetti A, Fortunato S, Kertész J. Detecting the overlapping and hierarchical community structure in complex networks. *New J Phys*, 2009, 11: 033015
- 15 Raghavan U N, Albert R, Kumara S. Near linear time algorithm to detect community structures in large-scale networks. *Phys Rev E*, 2007, 76: 036106
- 16 Gregory S. Finding overlapping communities in networks by label propagation. *New J Phys*, 2010, 12: 103018
- 17 Xie J, Szymanski B K. Towards linear time overlapping community detection in social networks. In: *Proceedings of the 16th Pacific-Asia Conference on Knowledge Discovery and Data Mining*, Kuala Lumpur, 2012. 25–36
- 18 Chen J, Wan Y. Research on label propagation algorithm based on modularity maximization in the social network. *J Commun*, 2017, 38: 25–33 [陈晶, 万云. 社交网络中基于模块度最大化的标签传播算法的研究. *通信学报*, 2017, 38: 25–33]
- 19 Li C Y, Tang Y, Lin H, et al. Parallel overlapping community detection algorithm in complex networks based on label propagation. *Sci Sin Inform*, 2016, 46: 212–227 [李春英, 汤庸, 林海, 等. 基于标签传播的可并行复杂网络重叠社区发现算法. *中国科学: 信息科学*, 2016, 46: 212–227]
- 20 Wu Z H, Lin Y F, Gregory S, et al. Balanced multi-label propagation for overlapping community detection in social networks. *J Comput Sci Technol*, 2012, 27: 468–479
- 21 Liu S H, Zhu F X, Gan L. A label-propagation-probability-based algorithm for overlapping community detection. *Chin J Comput*, 2016, 39: 717–729 [刘世超, 朱福喜, 甘琳. 基于标签传播概率的重叠社区发现算法. *计算机学报*, 2016, 39: 717–729]
- 22 Zhang C L, Wang Y L, Wu Y J, et al. Multi-label propagation algorithm for overlapping community discovery based on information entropy and local correlation. *J Chin Comput Syst*, 2016, 37: 1645–1650 [张昌理, 王一蕾, 吴英杰, 等. 基于信息熵和局部相关性的多标签传播重叠社区发现算法. *小型微型计算机系统*, 2016, 37: 1645–1650]
- 23 Xin Y, Yang J, Xie Z Q. A semantic overlapping community detection algorithm based on semantic data elds. *Sci Sin Inform*, 2015, 45: 918–933 [辛宇, 杨静, 谢志强. 基于数据场分析的语义重叠社区发现算法. *中国科学: 信息科学*, 2015, 45: 918–933]
- 24 Nagarajan N, Sen P, Chaoji V. Community detection in content-sharing social networks. In: *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, Niagara Falls, 2013. 82–89
- 25 Blei D M, Ng A Y, Jordan M I. Latent dirichlet allocation. *J Mach Learn Res*, 2003, 3: 993–1022
- 26 Duan L, Zhu X Y. Microblog community detection method based on community spatio-temporal topic model. *J Univ Electron Sci Tech China*, 2014, 43: 465–468 [段炼, 朱欣焰. 基于社区时空主题模型的微博社区发现方法. *电子科技大学学报*, 2014, 43: 465–468]
- 27 Yang J, Xin Y, Xie Z Q. Semantics social network community detection algorithm based on topic comprehensive factor analysis. *J Comput Res Dev*, 2014, 51: 559–569 [杨静, 辛宇, 谢志强. 基于话题综合因子分析的语义社会网络社区发现算法. *计算机研究与发展*, 2014, 51: 559–569]
- 28 Xin Y, Yang J, Xie Z Q. An overlapping semantic community structure detecting algorithm by label propagation. *Acta Autom Sin*, 2014, 40: 2262–2275 [辛宇, 杨静, 谢志强. 基于标签传播的语义重叠社区发现算法. *自动化学报*, 2014, 40: 2262–2275]
- 29 Xin Y, Yang J, Xie Z Q. An overlapping community structure detecting algorithm in semantic social network based on block field. *Acta Autom Sin*, 2015, 41: 362–375 [辛宇, 杨静, 谢志强. 一种面向语义重叠社区发现的 Block 场取样算法. *自动化学报*, 2015, 41: 362–375]

- 30 Xin Y, Yang J, Xie Z Q. A semantic overlapping community detecting algorithm in social networks based on random walk. *J Comput Res Dev*, 2015, 52: 499–511 [辛宇, 杨静, 谢志强. 基于随机游走的语义重叠社区发现算法. *计算机研究与发展*, 2015, 52: 499–511]
- 31 Xin Y, Yang J, Xie Z Q. Link-Block method for the semantic overlapping community detection. *J Softw*, 2016, 27: 363–380 [辛宇, 杨静, 谢志强. 一种面向语义重叠社区发现的 Link-Block 算法. *软件学报*, 2016, 27: 363–380]
- 32 Perozzi B, Al-Rfou R, Skiena S. Deepwalk: online learning of social representations. In: *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, 2014. 701–710
- 33 Grover A, Leskovec J. Node2vec: scalable feature learning for networks. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, 2016. 855–864
- 34 Han Z M, Tan X S, Chen Y, et al. NCSS: an effective and efficient complex network community detection algorithm. *Sci Sin Inform*, 2016, 46: 431–444 [韩忠明, 谭旭升, 陈炎, 等. NCSS: 一种快速有效的复杂网络社团划分算法. *中国科学: 信息科学*, 2016, 46: 431–444]
- 35 Newman M E J. Finding community structure in networks using the eigenvectors of matrices. *Phys Rev E*, 2006, 74: 036104
- 36 Gregory S. An algorithm to find overlapping community structure in networks. In: *Proceedings of European Conference on Principles and Practice of Knowledge Discovery in Databases*. Berlin: Springer, 2007. 91–102
- 37 Boguñá M, Pastor-Satorras R, Díaz-Guilera A, et al. Models of social networks based on social distance attachment. *Phys Rev E*, 2004, 70: 056122
- 38 Nicosia V, Mangioni G, Carchiolo V, et al. Extending the definition of modularity to directed graphs with overlapping communities. *J Stat Mech*, 2009, 2009: P03024

Optimal community detection method based on average mutual information

Dong LI¹, Mingquan CHENG¹, Yang XU^{1*}, Feng YUAN², Yinan CHEN¹ & Yaqing FU¹

1. *School of Software Engineering, South China University of Technology, Guangzhou 510006, China;*

2. *Institute of Software Application Technology, Guangzhou & Chinese Academy of Sciences, Guangzhou 511458, China*

* Corresponding author. E-mail: xuyang@scut.edu.cn

Abstract This study proposes an optimal community detection method based on average mutual information (AMI). We calculate the optimal community partition using AMI. This method is applied to the non-overlapping community detection algorithm (GN) and the overlapping community detection algorithm (COPRA), and the AMI-GN algorithm and the AMI-COPRA algorithm are generated respectively. Compared with the performance of the original GN, FN, and IE algorithm, experimental results show that AMI-GN algorithm improves the quality of community detection. Furthermore, compared with the performance of the original COPRA algorithm and the LPPB algorithm, experimental results show that the AMI-COPRA algorithm improves the stability of the original COPRA algorithm, reduces the average number of iterations, and accelerates the convergence speed of the algorithm. Moreover, compared with the LPPB algorithm, the AMI-COPRA algorithm reveals that the quality of the community shows a little difference, but is more stable than the LPPB algorithm. Our study shows that the AMI-based methods can improve the performance of typical non-overlapping community discovery algorithms and overlapping community discovery algorithms.

Keywords AMI-COPRA algorithm, AMI-GN algorithm, average mutual information, AMI method, community detection

Dong LI was born in 1970. He received his B.S. degree from Harbin Institute of Technology in 1992 and his Master's and Ph.D. degrees from Huazhong University of Science and Technology in 1995 and 2001, respectively. He is a Professor at South China University of Technology in China. His research interests include query optimization, big data analysis, and cloud computing.

Mingquan CHENG was born in 1994. He is an M.S. candidate at South China University of Technology. His research interests include social community detection.

Yang XU was born in 1970. He obtained his Ph.D. degree in computer software and theory from Wuhan University in 2007. He is a lecturer at South China University of Technology. His research interests are big data and cloud computing, process mining and modeling, and service computing.