考虑内外均衡安全增益的可量化社区隐藏算法

赵 霞[1] 魏霖静[1] 肖 君[2]

[1](甘肃农业大学信息科学技术学院 甘肃 兰州 730000)

[2](甘肃省计算中心系统集成部 甘肃 兰州 730000)

摘 要 为提高社区检测算法应对特定社区隐藏因素的能力,提出一种考虑内外均衡安全增益的可量化社区隐藏算法。通过研究社区隐藏的内在机制,推动社区检测算法性能的提升[1]。在给出社区网络模型的基础上,对社区隐藏的评价指标进行定义,实现了社区隐藏的量化分析;基于社区检测的安全性增益指标对社区隐藏过程中的节点添加和边缘的删除策略进行研究,基于这些操作实现对网络社区的更新;对社区检测隐藏算法的安全性进行了理论分析,为社区隐藏算法应用提供理论基础。通过在选取的4种社区网络实例中的仿真实验显示,该算法具有优异的社区隐藏性能和计算效率。

关键词 内外均衡 安全增益 可量化 社区隐藏

中图分类号 TP391 **文献标识码** A DOI:10.3969/j.issn.1000-386x.2018.12.052

QUANTIFIABLE COMMUNITY HIDING ALGORITHM CONSIDERING INTERNAL AND EXTERNAL EQUILIBRIUM SECURITY GAIN

Zhao Xia[1] Wei Linjing[1] Xiao Jun[2]

[1](*School of Information Science and Technology, Gansu Agricultural University, Lanzhou 730000, Gansu, China*)

[2](*Department of System Integration, Gansu Computing Center, Lanzhou 730000, Gansu, China*)

Abstract In order to improve the ability of community detection algorithm to deal with specific community hidden factors, this paper proposed a quantifiable community hiding algorithm considering internal and external equilibrium security gain. We tried to promote the performance of community detection algorithm by studying the inherent mechanism hidden in the community. On the basis of the model of community network, we defined the evaluation indicators hidden in the community and realized the quantitative analysis hidden in the community. The strategy of adding nodes and deleting edges in the process of community hiding was studied based on security gain index of community detection, and the network community was updated according to these operations. The security of community detection hiding algorithm was analyzed theoretically, which provided a theoretical basis for the application of community hiding algorithm. Simulation results in four selected community network instances show that the algorithm has excellent performance of community hiding and computational efficiency.

Keywords Internal and external equilibrium Security gain Quantifiable Community hiding

0 引 言

网络遍布于我们的日常生活中,网络的研究涉及到许多学科,从物理学到计算机、社会科学等诸多方面。网络分析中的一个重要任务是社区检测识别,即网络的区域(顶点子集)划分,以帮助其了解网络本身的结构和特征。

虽然社区检测研究非常广泛的课题,但很少有技术能够允许从社区检测算法中隐藏目标社区。本文研

收稿日期:2018-05-20。甘肃省科技厅2018年自然科学基金项目(18JR3RA79)。**赵霞**,讲师,主研领域:数据挖掘,电子商务。**魏霖静**,副教授。**肖君**,助理研究员。

究的目的是提供这个问题的深入解决策略,称之为社区隐藏。可以从两个不同的层面对这个问题进行研究。隐藏技术对于想在 Facebook 或 Twitter 等社交网络中隐藏(作为群体)的用户是有用的。有观点认为,想要隐藏的社区不应该是网络的一部分,而从隐私权的角度,很多网络中的个体节点希望不被网络检测工具检测到。研究社区隐藏的另一个意义是:针对恶意目的的社区检测攻击,可以研究其作用机理,制定出有效的防御机制。在研究社区隐藏过程中,存在下列需要解决的问题:(1) 隐藏模型问题。这里通过关注成员节点所拥有的真实的网络知识量将社区隐藏问题处理为优化问题。然后,引入社区安全性指标,以隐藏函数作为优化目标。研究表明,基于最优模块化的社区隐藏策略需要对社区结构具有先验知识,这取决于产生这些社区的社区检测算法[2]。基于安全性的社区隐藏不会受到这个问题的影响。(2) 如何量化特定目标社区的检测算法的欺骗水平。通过引入隐藏指标,给出这方面的形式定义。(3) 高效算法的设计。社区安全性指标的高值优化,某种方法可以搜索所有可能的候选,但可能计算效率不符合要求。

本文研究的目的是展示在各种真实网络上,通过不需要全局网络知识的近似优化技术,可以找到好的隐藏解决方案。该算法可扩展到具有数百万个节点和边缘的网络,并且比社区检测算法快得多,能够先于社区检测网络算法完成社区的隐藏操作[3]。

1　社区检测隐藏问题描述

本节将描述社区隐藏问题,并提供一个问题示例。首先从一些初步的定义开始。网络 $G=(V,E)$ 是一个无向图模型,其中 V 是网络模型的节点集,E 是网络模型的边集。

由社区检测算法 A_D 发现的一组社区(即,社区结构)可表示为:$\overline{C}=\{C_1,C_2,\cdots,C_k\}$,其中 $C_i\subseteq V$,$C_i\cap C_j=\varnothing$,$i,j\in\{1,2,\cdots,k\}$,$i\neq j$。给定社区 $\overline{V}\subseteq V$,社区 $\overline{V}$ 内边缘具有形式 $(u,v):\{u,v\}\in\overline{V}$[4]。

给定网络模型 $G=(V,E)$,目标社区可表示为 $C\subseteq V$,其为想要脱离社区的检测算法。如果 $C\in\overline{C}=\{C_1,C_2,\cdots,C_k\}$,对于最坏情形:目标社区已经完全被发现。另一方面,如果 $C\notin\overline{C}$,则 C 中的成员可以有不同的方式隐藏在 $\overline{C}$ 中。在引入隐藏之前,建立一些可将 C 良好隐藏在 $\overline{C}$ 中的必要条件:(1) 可达性保持,$C's$成员之间应该相互接触,以保持信息交流;(2) 社区传播,$C's$ 成员应该在尽可能多的社区 $\overline{C}$ 中进行传播;(3) 社区隐藏,$C's$ 成员应该分布在 $\overline{C}$ 中最大社区内。

这些需求表征社区隐藏评价指标,在正式定义隐藏评价指标之前,引入一些符号定义。给定社区结构 $\overline{C}=\{C_1,C_2,\cdots,C_k\}$,定义检测算法 A_D 的目标社区 C 的召回率指标如下:

$$R(C_i,C)=\frac{A_D(C's)}{|C|}\tag{1}$$

类似地,可定义精度指标为:

$$P(C_i,C)=\frac{A_D(C's)}{|C_i|}\tag{2}$$

式中:$\forall C_i\in\overline{C}$,$\forall C_i\cap C\neq\varnothing$,$A_D(C's)$ 为算法 A_D 检测到的 $C's$ 成员的数量。

定义 1　(隐藏指标)给定社区 C 和由社区检测算法发现的社区结构 $\overline{C}=\{C_1,C_2,\cdots,C_k\}$,社区隐藏分数可定义为:

$$H(C,\overline{C})=\left(1-\frac{|S(C)|-1}{|C|-1}\right)\times\frac{1}{2}\left(2-\max_{C_i\in\overline{C}}\{R(C_i,C)\}-\frac{\sum_{C_i\cap C\neq\varnothing}P(C_i,C)}{|C_i\cap C\neq\varnothing|}\right)\tag{3}$$

式中:$S(C)$ 是 $C's$ 成员在子图中连接的组件数量。H 通过定义 1 中的第一乘法因子捕获可达性存储(条件 1)。最好的情况是当所有节点都在一个单独的连接组件中时,相反,最坏的情况发生在它们都属于不同的连接组件时。社区传播(条件 2)由式(3)中的 $\max_{C_i\in\overline{C}}\{R(C_i,C)\}$ 捕获,其中包括最大召回率 R。社区隐藏(条件 3)由式(3)中的 $\frac{\sum_{C_i\cap C\neq\varnothing}P(C_i,C)}{|C_i\cap C\neq\varnothing|}$ 捕获,其基于平均精度指标 P。理想情况下,每个 $C_i\in C$ 包含少量百分比的 $C's$ 节点[5]。

总体而言,存在 $H1$,如果:(1) $C's$ 节点在单个连通分量中(满足条件 1);(2) C 中的每个成员属于不同社区(满足条件 2);(3) C 中的每个成员属于大社区(满足条件 3)。存在 $H0$,如果:(1) C 中的每个成员属于不同的组件;(2) $C\in\overline{C}$。下面将从计算的角度描述社区隐藏问题。

2　考虑安全增益的可量化社区隐藏

2.1　社区隐藏问题模型

隐藏分数 H 评估社区隐藏在社区结构 C 的水平,其中社区结构 $\overline{C}$ 可通过一些检测算法发现。本文的核心思想是通过仔细地重新排列 $C's$ 成员的 β 边来提高 C 的隐藏得分。

解决社区隐藏问题的直接方法是使用隐藏分数 H。然而,H 合并有关社区结构 $\overline{C}$,因此需要了解生成 $\overline{C}$ 的社区检测算法 A_D。在现实背景下,$C's$ 成员并不清楚社区结构 $\overline{C}$。因此,这里将检测算法视为黑箱。此外,为了制造 $C's$ 成员适用的隐藏,这里专注于不需要全局网络知识的算法。

定义 2 (社区隐藏)令网络模型为 $G=(V,E)$,给定目标社区 $C\in V$,以及更新的预算 β,则解决社区隐藏问题等于求解以下优化问题:

$$f=\underset{\{E'(C),\bar{E}'(C)\}}{\operatorname{argmax}}\{\phi(C,E(C),\bar{E}(C),\beta,E'(C),\bar{E}'(C))\} \quad (4)$$

式中:函数 $\phi(\cdot)$模型是一种社区隐藏算法;预算 β 限制了边缘更新的数量。隐藏函数 ϕ 与隐藏得分 H 的关键区别在于前者选择最大化 ϕ 的 β 变化,而 H 可量化目标社区 C 的理想属性(尽可能隐藏在检测算法的输出中)[6]。

2.2 社区检测隐藏算法框架

图 1 给出了社区隐藏算法的过程。社区隐藏算法的输入是社区 C 和更新 β 的预算。虽然社区检测算法需要全局的网络知识,而隐藏算法仅需知道 $C's$ 成员及其链接即可。作为社交网络的一部分,图 1 社区隐藏过程将 C 的成员暴露给社区检测过程,因为网络拓扑固有地包含关于社区成员资格的信息。这里把社区隐藏看作是来自 $C's$ 成员的共同努力,并根据隐藏函数 ϕ 重新定义 β 更新。C 内边缘的重新连接,包含边缘删除和边缘添加。这里给出具体示例,例如 Facebook 网络中,可以通过$C's$成员的"不结盟"来简单地实现边缘删除。而对于边缘添加,在 Facebook 中,需要接受友好请求。相反地,通过发现目标节点在同事、同学甚至是随机人之间的受欢迎程度,来实现边缘 C 的添加,这需要发现新的网络成员[7]。如何评估社区检测隐藏算法的主要思路是对比 $\overline{C}$ 中社区 C 的隐藏分数,初始社区结构,以及应用 β 更新后的社区结构$\overline{C}'$。

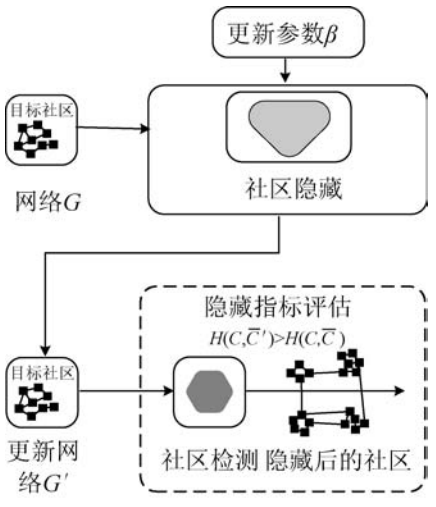


图 1 社区隐藏过程

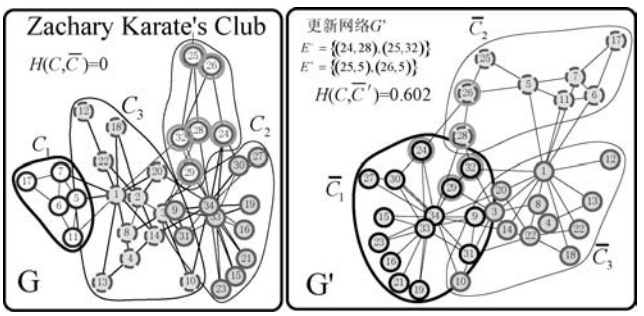
2.3 算法描述

本文的社区隐藏算法见算法 1。算法 1 只考虑了 C 内边缘添加,而不考虑 C 间边缘删除。并且,为选择最优更新,仅需获得 C 内边缘/节点度即可[8]。因此,算法所需的网络知识量是有限的。

算法 1 基于安全性的社区隐藏

```
1. procedure COMDECEPTSAFENESS(G=(V,E),C,β)
2. while β>0 && (ξ_C^add >0 ‖ ξ_C^del >0)
3. n_p = getNodeMinimumAddRatio(C);
4. n_t = findExteralNode(n_p,C,G);
5. ξ_C^add ←getAdditionGain(n_p,n_t,C);
6. (n_k,n_l)←getBestDelExclBridges(C);
7. ξ_C^del ←getDeletionGain(n_k,n_l,C);
8. if ξ_C^add ≥ ξ_C^del && ξ_C^add >0 then
9. G←(V,E∪{(n_p,n_t)});
10. else if ξ_C^del >0 then
11. G←(V,E\{(n_k,n_l)});
12. β=β-1;
13. endwhile
```

算法 1 中,第 3 行 getNodeMinimumAddRatio( ) 可对于每个 $n\in C$,计算在 C 以外的 n 个边缘的分数。第 4 行的 findExteralNode( ) 目的是找到边缘(n_p,n_t)不存在节点的。第 6 行 getBestDelExclBridges( ) 分两个阶段执行;获得最佳的边缘链接;选择最方便的边缘更新,并将其应用于网络。最佳的节点添加和边缘删除策略,主要是基于式(4)所示的优化目标函数进行执行,具体可通过以下实例进行描述,如图 2 所示。



(a) 空手道俱乐部网络 (b) 更新网络模型

图 2 算法应用实例

图 2(a)为测试对象。表 1 为考虑参数 $\beta=4$,目标社区 C 情况下,候选更新和实际更新选择。对于第一个节点的更新,该算法选择候选节点为 25 和 26,其位于跨社区的边缘之外,因为这些是具有最小比率的节点[9]。事实上,节点 25 和 26 都具有 0 个社区边缘。安全度增加 $\xi_C^{\text{add}}=0.125$。该算法还选取了边缘(24,28)作为社区内删除候选,因为这个边缘可最大化因子 $\frac{|\bar{E}(u,C)|}{2\deg(u)(\deg(u)-1)}-\frac{\bar{E}(u,C)}{2\deg(w)(\deg(w)-1)}$。值得

注意的是,边缘(29,32)没有被选为候选,是因为它是一个桥接,并且若它被删除将断开 C。边缘(24,28)的删除会带来 $\xi_C^{\text{del}}=0.358$ 的安全度增加。最后,因为 $\xi_C^{\text{del}}>\xi_C^{\text{add}}$ 应用的第一个更新是删除社区内边缘(24,28)。更新后的网络结构件如图 2(b)所示。更新计算过程见表 1。

表 1　网络更新计算过程

#更新	安全性			
	节点添加	ξ_C^{add}	边缘删除	ξ_C^{del}
1	(25,-)	0.125	(24,28)	0.358
2	(25,-)	0.125	(25,32)	0.25
3	(25,-)	0.125	--	0
4	(26,-)	0.125	--	0

对于给定社区 C 和参数 β,算法 1 对初始网络 G 进行更新,得到更新后的网络 G',满足 $\sigma(C')\geqslant\sigma(C)$。对于算法 1 所示的基于安全性的社区隐藏算法,最佳的节点添加和边缘删除策略,可通过检测 C 中的所有节点添加和边缘删除计算获得[10],其计算复杂度为 $O(|C|+|E(C)|)$。节点添加和边缘删除过程的计算复杂度为 $O(|E(C)|)$[11]。

3　社区检测隐藏算法的安全性分析

3.1　安全性定义

所提基于安全性的隐藏算法 D_s 的目标是通过重新连接 $C's$ 成员来隐藏检测算法。算法选择 $C\in\overline{C}$ 是一个很好的选择检测算法。这里重新链接算法是基于节点安全性进行设计的。给定社区 $\overline{V}\subseteq V$,利用 $E(u,\overline{V})$ 表示节点 $u\in\overline{V}$ 的内社区边缘集。

定义 3　(节点安全性)给定网络模型为 $G=(V,E)$,社区结构 $C\subseteq V$,以及社区 C 成员 $u\in C$。则 G 中 u 的安全性 $\sigma(u,C)$ 可定义为:

$$\sigma(u,C):=\frac{1}{2}\frac{|V_C^u|-|E(u,C)|}{|C|-1}+\frac{1}{2}\frac{|\tilde{E}(u,C)|}{\deg(u)}\quad(5)$$

式中:$V_C^u\subseteq C$ 是只通过 C 中节点从 u 可到达的节点集。式(5)右侧首项中,分子 $|V_C^u|-|E(u,C)|$ 用于计算 C 内在 u 中具有相同连接分量的节点数[12],以及连接 u 与其他成员 $C's$ 的边数之间的差值,其取值范围是 $[0,|C|-1]$。当 u 不能到达 C 中的任何节点时,可得其值为 0,即 $|V_C^u|=0$。对于另一极端情形,可得 $|C|-1$ 的值,如果:(1) C 可形成单个连通分量,即 $|V_C^u|=C$;(2) 只有一个连接 u 到 C 中节点的边,即 $|E(u,C)|=1$。对于分母,利用 $|C|-1$ 对 $|V_C^u|-|E(u,C)|$ 项进行归一化 $[0,1]$。综上,该项考虑了 C 中节点的一部分,节点只能通过 C 节点内的其他节点来平衡,这是由社区内边缘的数量来平衡的,并且给出了 u 可以在 C 中传输信息的程度说明。在理想情况下,C 成员将能够以最小数量的社区内边缘到达 C 的所有其他成员。

式(5)右侧次项中,给出 u 的边缘的一部分,并给出了 u 如何在网络内隐藏其度的解释。为了提高安全性,u 应该多样化其连接,即与不在 C 中的节点有较高比例的连接。同时,在区间 $[0,1]$ 中对式(5)和返回节点安全性值的两个分量进行加权。

定义 4　(社区安全性)给定网络模型为 $G=(V,E)$,社区结构 $C\subseteq V$,C 的安全性可定义为:$\sigma(C)=\sum_{u\in C}\frac{\sigma(u,C)}{|C|}$。

从 C 成员的安全性角度,允许识别最不安全的成员并重新连接其链接以增加整个 C 的安全性得分。安全性控制社区的不同属性有可达性和内部/外部边缘平衡。

3.2　边缘更新对安全性的影响

1) 边缘添加:令网络模型为 $G=(V,E)$,给定目标社区 $C\in V$,C 的安全性为 $\sigma(C)$。令 $\xi_C=\sigma(C')-\sigma(C)$ 为安全增益。

定理 1　对于任意的 C 内边缘添加 (u,w),其中 $u\in C,w\notin C$,则对于给定的 $G'=(V,E\cup\{(u,w)\})$:(1) $\xi_C\geqslant 0$ 始终满足;(2) 节点 $u\in\operatorname{argmin}\left\{\frac{|E(u,C)|}{\deg(u)}\right\}$ 的安全性增加幅度最大。

证明:检测如果满足 $\xi_C=\sigma(C')-\sigma(C)\geqslant 0$,则有:

$$\frac{1}{2}\frac{|\tilde{E}(u,C)|+1}{\deg(u)+1}-\frac{1}{2}\frac{\tilde{E}(u,C)}{\deg(u)}\geqslant 0\quad(6)$$

因为存在 $\deg(u)\geqslant\tilde{E}(u,C)$,则条件(1)成立,此外当条件(2)满足时,安全性增加幅度最大。

2) 边缘删除:本节主要从 C 内边缘进行边缘删除操作。

定理 2　C 间边缘 (u,w) 删除,其中 $u\in C,w\notin C$,则对于给定的 $G'=(V,E\setminus\{(u,w)\})$,$\xi_C\leqslant 0$ 始终成立。

证明:检测如果满足 $\xi_C=\sigma(C')-\sigma(C)\geqslant 0$,则有:

$$\frac{1}{2}\frac{|\tilde{E}(u,C)|-1}{\deg(u)-1}-\frac{1}{2}\frac{\tilde{E}(u,C)}{\deg(u)}\geqslant 0\quad(7)$$

因为 $\deg(u)\geqslant\tilde{E}(u,C)$,则上述定理成立。

定理 3　社区 C 内边缘 (u,w) 删除,并不总是带

来安全增益。

证明:假设删除边缘(u,w)之后,节点 u 和节点 w 在 C 中仍然保持的相同连通分量。需要满足下列条件:

$$\sum_{v\in C\setminus\{u,w\}}\sigma(v,C)+\frac{|V_C^u|-|E(u,C)|+1}{2(|C|-1)}+\frac{|\tilde{E}(u,C)|}{2(\deg(u)-1)}+\frac{|V_C^w|-|E(w,C)|+1}{2(|C|-1)}+\frac{|\tilde{E}(w,C)|}{2(\deg(w)-1)}>\sum_{v\in C\setminus\{u,w\}}\sigma(v,C)+\frac{|V_C^u|-|E(u,C)|}{2(|C|-1)}+\frac{|\tilde{E}(u,C)|}{2\deg(u)}+\frac{|V_C^w|-|E(w,C)|}{2(|C|-1)}+\frac{|\tilde{E}(w,C)|}{2\deg(w)}$$

从上述不等式可以很容易地检查到当边缘删除后,提高安全性增益的前提是$\frac{|\tilde{E}(u,C)|}{2\deg(u)(\deg(u)-1)}+\frac{|\tilde{E}(u,C)|}{2\deg(w)(\deg(w)-1)}$具有最大值。

4 实验分析

4.1 实验设置

实验硬件设置:CPU i7-6400K 3.0 GHz,内存大小为 16 GB RAM,操作系统为 Microsoft Windows 7 旗舰版[13]。考虑了基线算法,随机地选择更新的类型和边缘添加/删除的端点[14]。为了验证本文算法的有效性,这里选取 4 种社区检测算法,检验社区隐藏算法的效果:(1) Louvain 社区检测算法(Louv),一种社区检测的多级模块化优化算法,其计算复杂度为 $O(|V|\log|V|)$;(2) InfoMap 社区检测算法(Info),其返回一个社区结构,并为随机游走提供了最短的描述长度,其计算复杂度为 $O(|E|)$;(3) Edge-Betweeness 社区检测算法(Edge),为一个层次分解过程,其中边缘删除过程按照评估分值的下降顺序执行;(4) SpinGlass 社区检测算法(Spin),通过最大化加权社区聚类来实现图划分,基于三角分析策略实现对社区检测度量,其计算复杂度为 $O(|E|\log|V|)$。为了提高实验结果的稳定性,以下实验数据,为相同情形下运行上述 4 种社区检测算法的结果均值。

因为没有标准的基准来测试隐藏算法性能,这里设计了算法 2 中的方法进行评价。因为本文评价的是社区隐藏算法对于社区的隐藏程度,即它返回正确社区的能力与我们的目标是正交的。假设在最坏的情况下进行实验(第 3 行),即假设目标社区 C 被完全发现。对于具体的检测算法,通过查看社区分布可选择不同规模的目标社区。

算法 2 社区隐藏性分析

1. procedure EVALUATEDECEPTIONALGOG (β,A_D,D)
2. $\bar{C}\leftarrow A_D(G)$/ * 隐藏更新前社区结构 * /
3. $C\leftarrow$getTargetCommunity ($\bar{C}$);
4. $M_G(\bar{C})\leftarrow$getModularity ($\bar{C}$,G);
5. $\sigma(C)\leftarrow$getSafeness (C,G);
6. $H(C,\bar{G})\leftarrow$getDeception (C,G,$\bar{C}$);
7. $G'\leftarrow$applyDeception (G,C,β,D_x)/ * $x\in\{s,m,r,w\}$ * /;
8. $\bar{C}'\leftarrow A_D(G')$;
9. $M_{G'}(\bar{C}')\leftarrow$getModularity ($\overline{C'}$,$G'$)
10. $\sigma(C')\leftarrow$getSafeness (C,G');
11. $H(C,\bar{C}')\leftarrow$getDeception (C,$\overline{C'}$,G')

算法 2 第 4 -6 行可计算社区检测算法的模块性、安全性和隐藏评估得分,第 7 行给出的是更新后的网络 G'。算法第 8 行,在 G'上运行算法 1 计算社区检测社区的模块性、安全性和隐藏评估得分。上述评估算法的目的是调查:(1) 社区隐藏算法如何从检测算法中隐藏社区 C;(2) 如何设定参数 β;(3) 社区隐藏算法对社区结构的影响;(4) 社区隐藏和检测算法的运行时间。

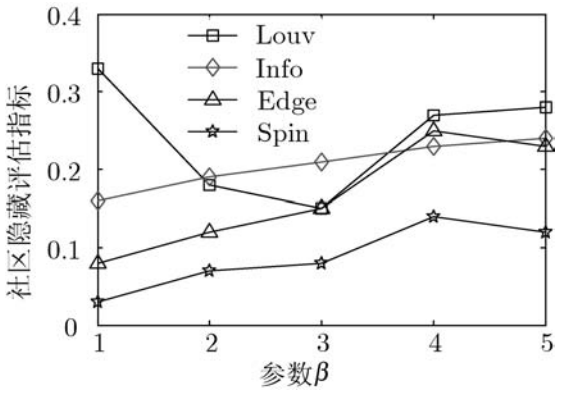
4.2 结果分析

本文选取的实验网络有 Zachary Karate's Club (kar)、Dolphins association(dol)、Madrid Train Bombing (mad)、Books about US politics (polb)4 种实验网络。表 2 给出了所考虑的网络的概述和选取的 4 种检测算法发现的社区数量。

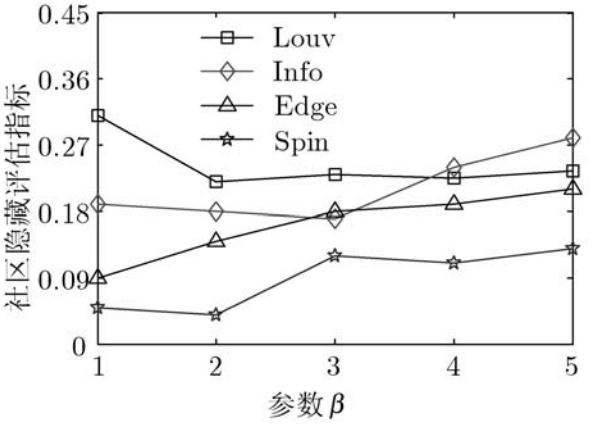
表 2 真实网络情形

网络	$\|V\|$	$\|E\|$	社区检测数量			
			Louv	Info	Edge	Spin
kar	34	78	4	3	4	5
dol	62	159	5	5.5	5	5
mad	62	159	6	7	4.5	--
polb	105	441	4	6	4	6

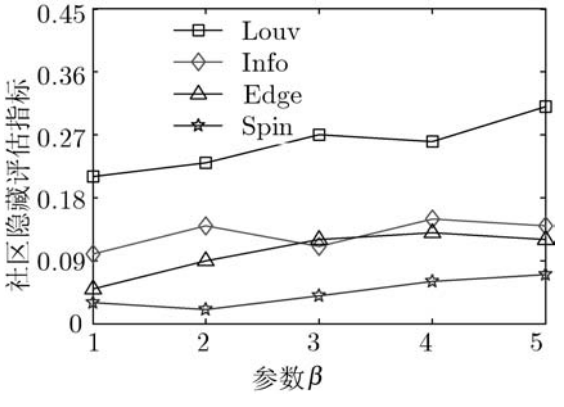
表 2 中,给出的 4 种检测算法发现的社区数量存在一定的差异,这与算法的检测性能相关。上述实验数据出现小数的原因是本文选取运行 20 次的结果均值计算结果。SpinGlass 社区检测算法在 mad 网络上的社区发现数量未得出,主要原因是该算法不能处理一定规模以上的网络。针对表 2 所示实验网络,参数 β 的取值区间是 1 ~5,图 3 给出实验所得的社区隐藏指标实验结果。



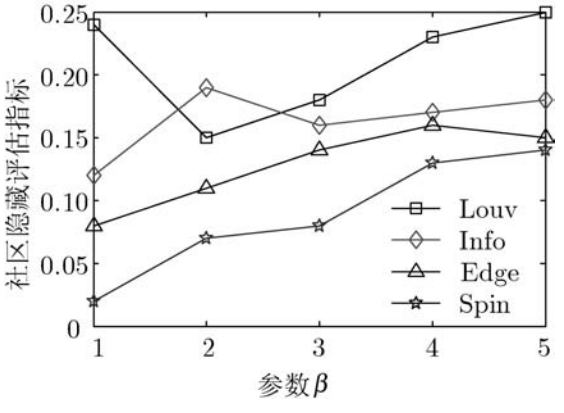
(a) kar 网络实验结果



(b) dol 网络实验结果



(c) polb 网络实验结果



(d) mad 网络实验结果

图 3　网络隐藏指标

根据图 3 所示网络隐藏指标实验结果可知,随着参数 β 的增大,几种算法在 4 种网络上的实验结果均呈现出增大的趋势。同时,可看出个别情形下呈现出波动性,对这样的行为可解释为:这些算法不是基于模块最大化的,算法产生一个更新网络,具有较低的模块化价值不会带来足够的中断 INF 的功能,产生更大的评估价值[15]。

此外,为更加直接地验证所提算法的社区隐藏性能,这里选取参数 β 取值为 1 ~5,验证算法在 kar、dol、mad、polb 4 种实验网络上,更新前、更新后以及随机节点边缘调整方式的特定社区检测概率。该结果为每种算法运行 30 次的均值结果,见表 3。

表 3　特定社区的隐藏性能

网络	算法	检测率/%	计算时间/s
kar	更新前	95.62	8.65
	本文算法更新	2.23	2.74
	随机算法更新	52.30	12.63
dol	更新前	94.69	7.98
	本文算法更新	3.84	2.51
	随机算法更新	61.50	10.82
mad	更新前	93.58	8.52
	本文算法更新	2.86	3.16
	随机算法更新	59.70	11.59
polb	更新前	94.19	6.97
	本文算法更新	3.46	2.92
	随机算法更新	62.80	9.87

表 3 所选取的特定社区选取标准是根据上述实验数据网络的真实社区划分结果选择的。其中本文方法和随机调整方式的节点和边缘更新数量均设定为 5。根据表 3 实验结果可知,在特定隐藏社区的检测概率上,本文算法更新后的网络始终保持在较低的水平上,在 3% 左右。而更新前网络特定隐藏社区的隐藏性能相对较差,都达到了 90% 以上,随机更新策略的社区检测率也达到了 50% 以上,这体现出所提算法较高的社区隐藏能力。在算法的执行时间指标上,本文算法更新后的网络的运行时间 2 ~4 s,而更新前网络的运行时间相对较长,这表明本文算法具有相对较高的社区隐藏效率,能够先于社区检测算法实现对社区的实时隐藏。

5　结　语

本文提出一种考虑内外均衡安全增益的可量化社

区隐藏算法,在给出社区网络模型的基础上,对社区隐藏的评价指标进行定义,实现了社区隐藏的量化分析。然后基于社区检测的安全性增益指标对社区隐藏过程中的节点添加和边缘的删除策略进行研究。基于这些操作实现对网络社区的更新,最后对社区检测隐藏算法的安全性进行了理论分析,为社区隐藏算法应用提供理论基础。本文研究目的并不是如何获得更佳的社区隐藏性能,而是通过研究社区隐藏过程,探索其存在的共性因素,从而对社区检测算法起到更多的指导意义。下一步的研究重点是:探索自然网络中存在的有意或者无意的社区隐藏行为,分析其存在的物理意义,并有针对性的实现检测突破,这对于提升社区检测算法性能具有重要的意义。

参 考 文 献

[1] Liu T, Tao D. On the performance of manhattan nonnegative matrix factorization[J]. IEEE Transactions on Neural Networks & Learning Systems, 2017, 27(9):1851 - 1863.

[2] 邹丹, 窦勇, 郭松. 基于 GPU 的稀疏矩阵 Cholesky 分解[J]. 计算机学报, 2014, 37(7): 1445 - 1454.

[3] Li J, Bioucas-Dias J M, Plaza A, et al. Robust collaborative nonnegative matrix factorization for hyperspectra unmixing (R-CONMF)[C]// The Workshop on Hyperspectral Image & Signal Processing: Evolution in Remote Sensing. IEEE, 2017:1 - 4.

[4] 蒋盛益, 杨博, 王连喜. 一种基于增量式谱聚类的动态社区自适应发现算法[J]. 自动化学报, 2015, 41(12): 2017 - 2025.

[5] 刘海洋, 王志海, 黄丹, 等. 基于评分矩阵局部低秩假设的成列协同排名算法[J]. 软件学报, 2015, 26(11): 2981 - 2993.

[6] Wang S S, Chern A, Yu T, et al. Wavelet speech enhancement based on nonnegative matrix factorization[J]. IEEE Signal Processing Letters, 2016, 23(8):1101 - 1105.

[7] Kitamura D, Ono N, Sawada H, et al. Determined blind source separation unifying independent vector analysis and nonnegative matrix factorization[J]. IEEE/ACM Transactions on Audio Speech & Language Processing, 2016, 24(9):1626 - 1641.

[8] Shi L, Zhang L, Zhao L, et al. Adaptive laplacian eigenmap-based dimension reduction for ocean target discrimination[J]. IEEE Geoscience & Remote Sensing Letters, 2016, 13(7):902 - 906.

[9] Venkataraman A, Yang D, Pelphrey K, et al. Bayesian community detection in the space of group-level functional differences[J]. IEEE Transactions on Medical Imaging, 2016, 35(8):1866 - 1882.

[10] Sammarco M, Campista M E M, Amorim M D D. Scalable wireless traffic capture through community detection and trace similarity[J]. IEEE Transactions on Mobile Computing, 2016, 15(7):1757 - 1769.

[11] 罗会兰, 万成涛, 孔繁胜. 基于 KL 散度及多尺度融合的显著性区域检测算法[J]. 电子与信息学报, 2016, 38(7): 1594 - 1601.

[12] Zhang X, Zong L, Liu X, et al. Constrained clustering with nonnegative matrix factorization[J]. IEEE Transactions on Neural Networks & Learning Systems, 2017, 27(7):1514 - 1526.

[13] 李艳雄, 吴水, 贺前华. 基于特征均值距离的短语音段说话人聚类算法[J]. 电子与信息学报, 2012, 34(6): 1404 - 1407.

[14] Kim Y D, Choi S. Variational bayesian view of weighted trace norm regularization for matrix factorization[J]. IEEE Signal Processing Letters, 2013, 20(3):261 - 264.

[15] Dutta P, Halder A, Bhattacharya R. Nonlinear estimation with perron-frobenius operator and karhunen-loève expansion[J]. IEEE Transactions on Aerospace & Electronic Systems, 2016, 51(4):3210 - 3225.

(上接第 273 页)

[6] 陈嘉霖,段家华,张明宇. 邻域粗糙集与相关向量机相结合的变压器故障综合诊断模型[J]. 电力系统及其自动化学报,2016,28(11):117 - 122.

[7] 曹付元,梁吉业,钱宇华. 基于信息熵的决策表约简[J]. 计算机应用,2005,25(11):2630 - 2631.

[8] 王国胤,于洪,杨大春. 基于条件信息熵的决策表约简[J]. 计算机学报,2002,25(7):759 - 766.

[9] 黄国顺,文翰. 基于边界域的条件信息熵和属性约简[J]. 计算机应用,2015,35(10):2771 - 2776.

[10] 李少年,吴良刚. 基于邻域信息熵度量数值属性快速约简算法[J]. 计算机工程与科学,2016,38(2):350 - 355.

[11] 王长宝,杨习贝,窦慧莉,等. 邻域决策错误率的局部约简方法研究[J]. 计算机工程与应用,2018(6):95 - 99,122.

[12] 何松华,康婵娟,鲁敏,等. 基于邻域组合测度的属性约简方法[J]. 控制与决策,2016,31(7):1225 - 1230.

[13] Hu Q, Yu D, Liu J, et al. Neighborhood rough set based heterogeneous feature subset selection[J]. Information Sciences An International Journal,2008,178(18):3577 - 3594.

[14] Teng S H, Lu M, Yang A F, et al. Efficient attribute reduction from the viewpoint of discernibility[J]. Information Sciences An International Journal, 2016, 326(C):297 - 314.

[15] Hu Q, Yu D, Liu J, et al. Neighborhood rough set based heterogeneous feature subset selection[J]. Information Sciences An International Journal,2008,178(18):3577 - 3594.

[16] Zhao H, Qin K. Mixed feature selection in incomplete decision table[J]. Knowledge-Based Systems, 2014, 57(2): 181 - 190.