



大规模社会网络 K-出入度匿名方法

张晓琳¹, 刘 娇¹, 毕红净², 李 健¹, 王永平¹

(1. 内蒙古科技大学 信息工程学院, 内蒙古 包头 014010; 2. 唐山师范学院 计算机科学系, 河北 唐山 063000)

摘 要: 现有社会网络隐私保护技术在处理大规模社会网络有向图时数据处理效率较低, 且匿名数据发布通常不能满足社区结构分析的需求。为此, 提出一种基于层次社区结构的大规模社会网络 K-出入度匿名(KIODA)算法。该算法基于层次社区结构划分社区, 采用贪心算法分组并匿名 K-出入度序列, 分布式并行添加虚拟节点以实现 K-出入度匿名, 基于 GraphX 图数据处理平台传递节点间的信息, 根据层次社区熵的变化情况选择虚拟节点对并进行合并删除, 从而减少信息损失。实验结果表明, KIODA 算法在处理大规模社会网络有向图数据时具有较高的执行效率, 并在匿名后保证了数据发布时社区结构分析结果的可用性。

关键词: 层次社区结构; 社会网络有向图; K-出入度匿名; 社区划分; GraphX 框架

开放科学(资源服务)标志码(OSID):



中文引用格式: 张晓琳, 刘娇, 毕红净, 等. 大规模社会网络 K-出入度匿名方法[J]. 计算机工程, 2020, 46(11): 164-173.

英文引用格式: ZHANG Xiaolin, LIU Jiao, BI Hongjing, et al. K-in&out-degree anonymity method for large scale social networks[J]. Computer Engineering, 2020, 46(11): 164-173.

K-In&Out-Degree Anonymity Method for Large Scale Social Networks

ZHANG Xiaolin¹, LIU Jiao¹, BI Hongjing², LI Jian¹, WANG Yongping¹

(1. School of Information Engineering, Inner Mongolia University of Science and Technology, Baotou, Inner Mongolia 014010, China;
2. Department of Computer Science, Tangshan Normal University, Tangshan, Hebei 063000, China)

[Abstract] Existing privacy protection techniques are inefficient when applied to directed graphs of large-scale social networks, and publishing anonymous data does not meet the needs of community structure analysis. To address the problem, this paper proposes a K-In&Out-Degree Anonymity(KIODA) algorithm for large-scale social networks based on hierarchical community structure. The algorithm divides the community based on hierarchical community structure. The greedy algorithm is used to group K-in&out-degree sequences and make them anonymous, and the virtual nodes are added in parallel to achieve K-in&out-degree anonymity. Then information exchanges between nodes are implemented based on the GraphX platform. Virtual node pairs are selected based on the changes of the hierarchical community entropy, and are merged and deleted to reduce information loss. Experimental results show that the KIODA algorithm improves the efficiency of processing directed graphs of large-scale social networks, and ensures the availability of community structure analysis results in data publishing after the anonymity is realized.

[Key words] hierarchical community structure; directed graph of social networks; K-In&Out-Degree Anonymity(KIODA); community division; GraphX framework

DOI:10.19678/j.issn.1000-3428.0056038

0 概述

随着社会网络的发展以及各种社交 App 的快速普及, 网络用户人数不断增加。目前, Facebook 用户

总数约为 21.7 亿, 微信用户总数约为 9.8 亿, 截至 2018 年 3 月 31 日, 支付宝为约 8.7 亿位活跃用户提供服务。用户在使用社会网络的过程中产生了大量社会网络数据, 对于实际社会网络有向图, 一些具有

基金项目: 国家自然科学基金“面向云计算环境的大规模社会网络隐私保护技术研究”(61562065); 内蒙古自治区自然科学基金“有效保护社区结构的大规模社会网络隐私保护技术研究”(2019MS06001)。

作者简介: 张晓琳(1966—), 女, 教授, 主研方向为大规模社会网络隐私保护; 刘 娇, 硕士研究生; 毕红净, 讲师、硕士; 李 健, 硕士研究生; 王永平, 讲师。

收稿日期: 2019-09-18 修回日期: 2019-11-06 E-mail: 2784899426@qq.com

相同爱好或者相似属性的用户还会形成各种特定的群体,即社会网络的社区结构。因此,对大规模社会网络有向图发布时的社区结构进行分析具有重要意义。文献[1]指出由于社会网络具有结构特征,简单的删除标识属性不能防止攻击者通过其他背景知识识别出目标用户。因此,研究人员针对不同的隐私信息提出了不同的隐私保护模型,如图修改模型^[2-4]、聚类泛化模型^[5-6]、数据扰动模型^[7]和差分隐私模型^[8]等。但是,目前大部分隐私保护技术仅针对社会网络无向图来保护个人隐私信息,忽略了对社会网络有向图社区结构的保护。

为满足 K-出入度匿名并有效保护大规模社会网络有向图的社区结构,本文基于 GraphX 图数据处理平台,建立基于层次社区结构的大规模社会网络 K-出入度匿名模型,并提出一种 K-出入度匿名(KIODA)算法。

1 相关工作

目前,社会网络隐私保护信息主要包括节点隐私、边隐私和图性质隐私 3 种^[9]。针对不同的社会网络隐私保护信息,文献[7]提出一种新的匿名方法,即拆分匿名化,通过拆分匿名处理的社交网络可以抵御直接攻击。文献[10]提出无向图的 k-度匿名概念,其要求图中任何节点都至少有 $k-1$ 个节点与其度数相同,使用贪心策略增加边来实现匿名,从而抵御节点度属性攻击。文献[11]提出一种 UMGA 算法,其采用贪心算法和穷举法生成匿名度序列,通过随机边选择和邻居中心性边选择方法修改无向图从而实现 k-度匿名。文献[12]提出一种改进的 K-度匿名算法,该算法保留了个人隐私信息以及社交网络结构属性。文献[13]提出一种图形结构感知的分层 k-匿名技术,其根据幂律分布的特征划分节点,在隐私保护过程中分析图形的结构特征,根据图形结构特征调整边缘。文献[14]提出一种保持网络结构稳定性的 k-度匿名隐私保护模型 SimilarGraph,该模型运用动态规划方法对社会网络按照节点度序列进行最优簇划分,通过移动边操作方式重构网络图以实现图的 k-度匿名化。文献[15]针对有向图提出了可达性保持图匿名化方法,通过图修改来防止隐私泄露,同时保证匿名图在社会网络分析和图查询方面的数据可用性。文献[16]提出一种保护链接关系的分布式匿名方法 PLRD-(k, m),其将互为 N-hop 邻居的节点分为一组并进行 k-度匿名和 m-标签匿名,保证攻击者无法通过度和标签识别出目标并保护链接关系不被泄露。现有的隐私保护方法虽然减少了图修改的信息损失,但改变了原始图的网络连通性,在匿名过程中没有考虑社会网络图的社区结构,从而影响了社区结构的性质分析并降低了

所发布数据的使用价值。

在进行大规模社会网络隐私保护的同时保护社会网络图的社区结构成为国内外学者关注的热点。文献[17]在保持总体图谱不变的前提下,通过节点之间相似度的比较来随机增删 k 条边。文献[18]利用基于图分割理论的社区划分算法计算拉普拉斯矩阵,设置谱约束条件,从而计算增删边的约束。文献[19]提出一种局部扰动的 k-匿名技术,在节点分组时使属于相同社区的节点分到一个组内,根据节点相似性重构社会网络无向图。文献[20]利用粗糙集的上近似概念划分社区并进行匿名,匿名前后保持图的社区结构性质。文献[21]提出一种新的边缘修改技术,对图表进行匿名化时较好地保留了图形社区结构,匿名后社会网络 K 核数保持不变,即保持社区结构不变。

目前,多数社会网络隐私保护方法存在处理大规模社会网络有向图时数据效率低、忽略了对社会网络有向图社区结构进行保护的问题,因此,本文提出一种基于层次社区结构的 KIODA 算法,以提高数据处理效率并保证数据发布时社区结构分析结果的可用性。

2 预备知识与问题定义

将社会网络表示为有向图 $G=(V, E)$, 其中, $V(G)$ 和 $E(G)$ 分别表示图 G 的节点集合和边集合。边 $\langle u, v \rangle$ 表示从节点 u 指向 v 的一条边,边 $\langle u, v \rangle$ 称为 u 的出边、 v 的入边,节点 u 的入边数目是 u 的入度数,记作 $d_{in}(u)$,节点 u 的出边数目是 u 的出度数,记作 $d_{out}(u)$,节点 u 的出入度用 $(d_{in}(u), d_{out}(u))$ 表示。

2.1 基于层次社区结构的社区划分

定义 1(有向图层次社区树 H_G) 给定一个社会网络有向图 G ,用层次随机图 HRG 来表示有向图 G 的社区结构,记作有向图层次社区树 H_G 。 H_G 的叶子节点表示有向图 G 中的节点, H_G 中的每个内部节点 r 代表叶子节点的连接概率 P_r , T_r 是 H_G 的一个内部节点 r 的子树,左子树 T_r^L 中的叶子节点与右子树 T_r^R 中的叶子节点在有向图 G 中的连接概率用 P_r 表示,其反映左子树与右子树叶子的联系强度, P_r 越大,节点联系越紧密。 P_r 的计算公式为:

$$P_r = \frac{|E_r|}{|T_r^R| \cdot |T_r^L| \cdot 2} \quad (1)$$

社会网络有向图 G 的 HRG 不是唯一的,因此,选择最优的 HRG,进而根据有向图层次社区树 H_G 得到社会网络有向图的社区结构。对于不同的 HRG,可以采用似然函数 L 来评价其对社会网络有向图 G 的合适程度。

定义 2(似然函数 L) 似然函数 L 是社会网络有向图 G 产生 H_G 的后验函数,选取 L 值最大的 HRG 树作为有向图 G 的 H_G 。 L 的计算公式为:

$$L(H_G) = \prod_{r \in H_G} [P_r^{p_r} (1 - P_r)^{1-p_r}]^{|T_r^R| \cdot |T_r^L| \cdot 2} \quad (2)$$

图 1(a)所示为社会网络有向图 G_0 ,图 1(b)~图 1(d)为社会网络有向图 G_0 的 3 个可能 HRG。

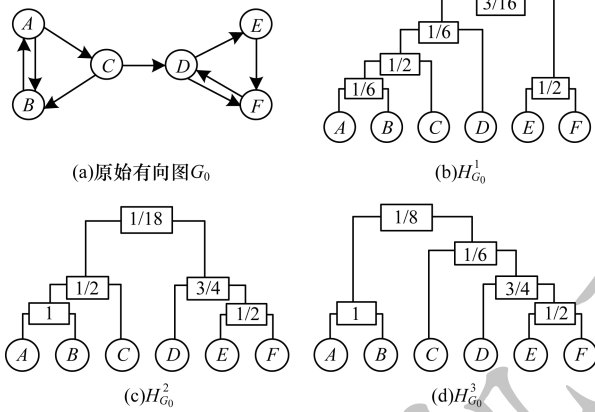


图 1 G_0 的社区划分示意图

Fig. 1 Community division schematic diagram of G_0

对有向图 G_0 列出 3 个可能的 HRG,计算似然函数 L 的值。经计算,图 1(b)的 $L = 4.639e^{-7}$,图 1(c)的 $L = 6.465e^{-5}$,图 1(d)的 $L = 4.255e^{-6}$,其中,图 1(c)的 L 值最大,因此,选择 H_G^2 作为有向图 G_0 的 H_G ,社区划分结果如图 2 所示。

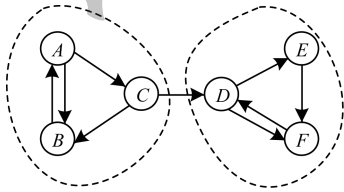


图 2 有向图 G_0 的社区划分结果

Fig. 2 Community division results of directed graph G_0

2.2 大规模社会网络 K-出入度匿名模型

在社会网络无向图中,攻击者可以通过度攻击识别出目标节点,但是对于社会网络有向图,节点具有出度和入度数。因此,本文构建一种出入度攻击模型。

定义 3(出入度攻击) 假设攻击者知道目标节点的入(出)度数,或者同时知道入度和出度数,攻击者通过这些背景知识能够识别出唯一目标节点,这种攻击被称为出入度攻击。

如图 3 所示,假设攻击者知道目标节点的出度数为 2,可以识别出 Alice、Kayla 和 Bob。如果攻击

者还知道目标节点的入度数为 1,则能识别出唯一目标节点 Alice,导致 Alice 节点的隐私信息泄露。

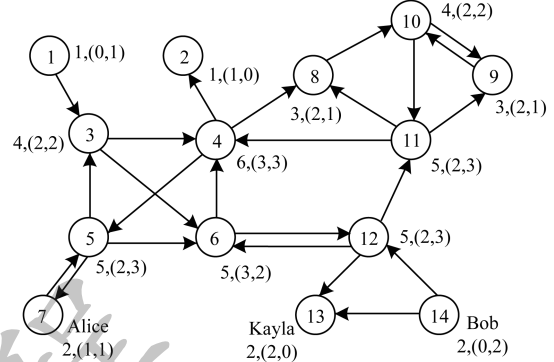


图 3 社会网络有向图 G

Fig. 3 Social network directed graph G

传统的 K-度匿名技术仅针对社会网络无向图,忽略了边的方向性,但是实际的社会网络图中的边存在方向,因此,本文针对社会网络有向图抵御出入度攻击,构造 K-出入度序列进行 K-出入度匿名。

定义 4(K-出入度匿名) 给定社会网络有向图 $G = (V, E)$ 和正整数 K ,对于有向图中任意节点 $v \in V(G)$,均存在 $m(m \geq K-1)$ 个其他节点与节点 v 的入度和出度数相等,即 $d_{in}(v) = d_{in}(v_i)$, $d_{out}(v) = d_{out}(v_i)$ ($1 \leq i \leq m$),则称有向图 G 为 K-出入度匿名图。

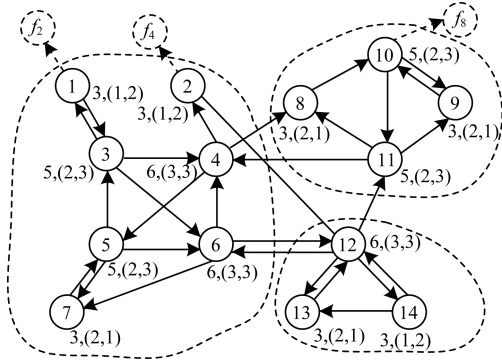
定义 5(基于层次社区结构的大规模社会网络 K-出入度匿名模型) 给定社会网络有向图 $G = (V, E)$ 和正整数 K 。根据如下 3 个步骤得到的匿名社会网络有向图 $G^* = (V^*, E^*)$,即符合基于层次社区结构的大规模社会网络 K-出入度匿名模型:

1) 基于层次社区结构划分社区,确定原始社会网络有向图 G 中节点所属的社区。

2) 对社会网络有向图 G 中节点的出入度序列进行排序、分组并匿名,得到 K-出入度匿名序列。

3) 根据 K-出入度匿名序列分布并行构造匿名图,基于 GraphX 传递节点间信息并迭代进行虚拟节点对合并与删除,提高数据发布时社区结构分析结果的可用性。

如图 4 所示,有向图 G^* 为社会网络有向图 G 的 3-出入度匿名图,其中, $K=3$,节点 1~节点 7 属于社区 1,节点 8~节点 11 属于社区 2,节点 12~节点 14 属于社区 3。对于任意节点 v ,均存在至少 2 个节点与其具有相同的入度和出度数,即攻击者不能以大于 $1/3$ 的概率唯一识别出目标节点,从而满足了 K-出入度匿名并保证了数据发布时社区结构分析结果的高可用性。

图4 匿名社会网络有向图 G^* Fig. 4 Anonymous social network directed graph G^*

3 K-出入度匿名算法

基于层次社区结构的大规模社会网络 K-出入度匿名算法 KIODA 结合了为大规模数据处理而设计的计算引擎 Spark, 算法在分布式并行环境下执行, 对大规模社会网络数据实现并行高效的隐私保护。

3.1 基于层次社区结构的社区划分算法

基于层次社区结构的社区划分算法首先得到社会网络有向图的 HRG, 然后借助马尔科夫蒙特卡洛抽样方法^[22]收敛筛选出较优的层次随机图 H_G , 进而得到社会网络图的社区结构。HRG 生成算法描述如下:

算法 1 Construst_HRG(G, ε_1) 算法

输入 原始图 G , 差分隐私参数 ε_1

输出 H_G

1. 根据 Markov chain 随机生成原始图 G 的一个 HRG, 作为 T_0 ;

2. $t \leftarrow 1$;

3. while Markov chain is not converged do

4. Randomly select an internal node n in T_{t-1}

5. Generate HRG T' by randomly transforming adjacent subtrees of node n

// 通过随机交换节点 n 的相邻 HRG 生成 T'

6. if the probability of transformation satisfies

$$\min \left(\frac{\exp\left(\frac{\varepsilon_1}{2\Delta u} \log_a L(T')\right)}{\exp\left(\frac{\varepsilon_1}{2\Delta u} \log_a L(T_{t-1})\right)}, 1 \right) \text{ then}$$

7. $T_t = T'$;

8. else

9. $T_t = T_{t-1}$;

10. end if

11. $t = t + 1$;

12. end while

13. return $H_G = T_t$;

算法 1 随机选择一个 HRG 来初始化马尔科夫链, 第 3 行 ~ 第 12 行多次迭代直到马尔科夫链收敛。设马尔科夫链的当前状态为 T_{t-1} , 随机选择一个内部节点 n , 替换 n 的相邻 HRG 产生下一个状态 T' 。由于选择具有随机性, T' 不唯一, 通过计算似然函数 L 的值筛选较优的 T' 。第 6 行 ~ 第 10 行通过比较交换前后

L 的误差值, 借助马尔科夫蒙特卡洛抽样方法, 设置状

态交换接受率为 $\min \left(\frac{\exp\left(\frac{\varepsilon_1}{2\Delta u} \log_a L(T')\right)}{\exp\left(\frac{\varepsilon_1}{2\Delta u} \log_a L(T_{t-1})\right)}, 1 \right)$ 。迭代

结束, 马尔科夫链收敛, 得到较优的 H_G 。图 3 所示的有向图 G 生成的较优 H_G 如图 5(a) 所示, 社区划分结果如图 5(b) 所示。

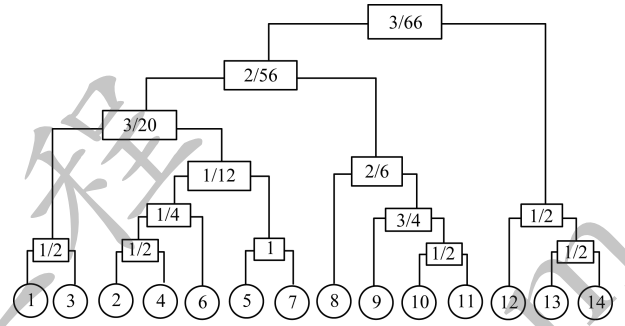
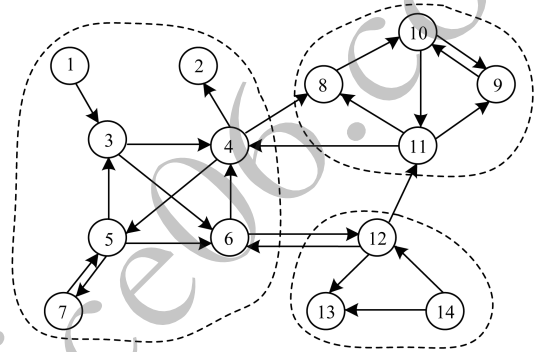
(a) 较优的 H_G (b) 图 G 的社区划分结果

图5 基于层次结构的社区划分示意图

Fig. 5 Community division schematic diagram based on hierarchical structure

3.2 K-出入度序列分组匿名算法

针对社会网络有向图, 本文提出 Sequence Partition(G, k) 算法, 同一组中目标出入度的入(出)度数为该分组中所有入(出)度数的最大值, 即 $\text{goal}(\text{in_deg}, \text{out_deg}) = (\max\{\text{分组中所有元素的 in_deg}\}, \max\{\text{分组中所有元素的 out_deg}\})$ 。K-出入度序列分组匿名算法描述如下:

算法 2 Sequence Partition(G, k) 算法

输入 原始有向图 G , 匿名参数 k

输出 K-出入度匿名序列 d^*

1. MF_Seq = [];

2. for each node $\langle \text{in_deg}, \text{out_deg} \rangle$ do

3. Insert in MF_Seq;

4. end for

5. Sort(MF_Seq) by $\langle \text{in_deg}, \text{out_deg} \rangle$;

6. last_partition_index = 0;

7. for $v_i \in \text{MF_Seq}$ do

8. for $l = \text{last}$ to $i - 1$ do

9. $\text{goal}_l(\text{in_deg}, \text{out_deg}) = \max(\text{in_deg}[i], \text{out_deg}[i])$;

```

10. end for
11. for m = i to i + k do
12. goal_2(in_deg, out_deg) = max(in_deg[m + 1], out_deg[m + 1]);
13. goal_3(in_deg, out_deg) = max(in_deg[m], out_deg[1]);
14. end for
15. SPC1 = goal_1 - MF_Seq[i], SPC2 = 0;
16. for j = i + 1 to i + k do
17. SPC1 = SPC1 + goal_3 - deg[j - 1];
18. SPC2 = SPC2 + goal_2 - deg[j];
19. end for
20. if SPC2 < SPC1 then
21. last_partition_index = i;
22. i = i + k;
23. else
24. i++;
25. end if
26. end for

```

算法 2 第 2 行 ~ 第 14 行根据节点出入度进行排序, 求出目标出入度。第 16 行 ~ 第 25 行判断 SPC1、SPC2 值的大小并进行分组。其中, SPC1 表示将当前元素合并到上一组中的匿名化成本, SPC2 表示将当前元素和后面 $k-1$ 个元素形成新组的成本。如果 $SPC2 < SPC1$, 该元素将和后面 $k-1$ 个元素形成一个新组, 反之则合并到上一组。原始图 G 的分组匿名过程如图 6 所示。

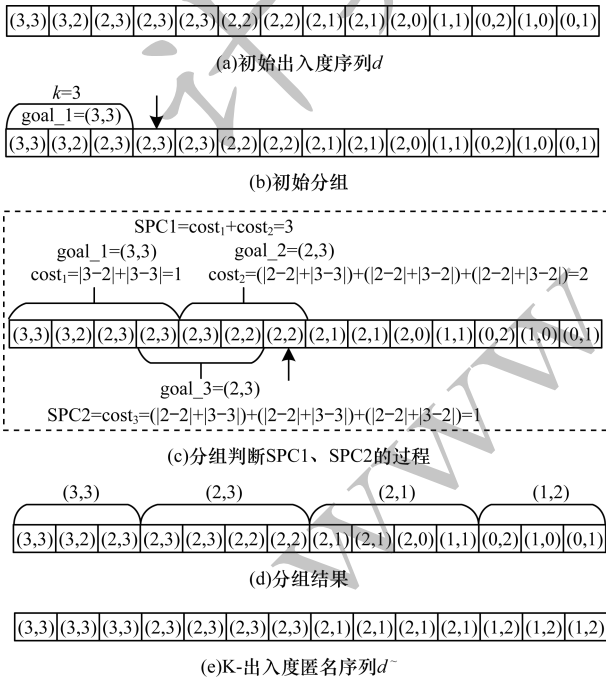


图 6 分组匿名过程

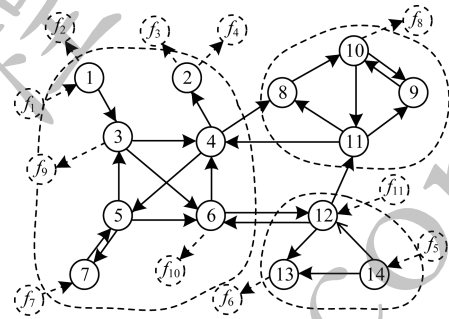
Fig. 6 Group anonymity process

图 6(a) 表示初始出入度序列 d 。假设 $k=3$, 前 3 个元素先被放入一个组中, 如图 6(b) 所示。判断第 4 个元素的分组情况, 如图 6(c) 所示, 此时 $goal_1 =$

$(3,3)$, $goal_2 = (2,3)$, $goal_3 = (2,3)$, $SPC1 = cost_1 + cost_2 = 3$, $SPC2 = cost_3 = 1$ 。 $SPC1 > SPC2$, 因此, 第 4 个元素和后面 2 个元素形成新组, 再判断第 7 个元素的分组情况。最终分组结果如图 6(d) 所示, 得到的 K-出入度匿名序列 d' 如图 6(e) 所示。

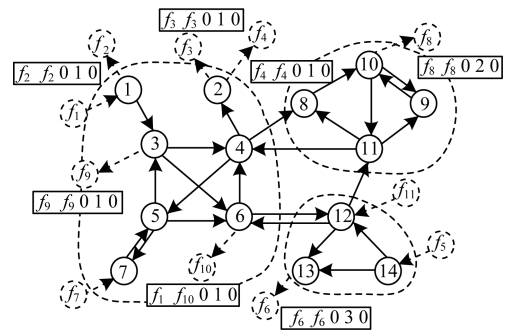
3.3 选择虚拟节点对合并删除算法

根据 K-出入度匿名序列分布并行构造匿名图, 添加虚拟节点得到图 7 所示的匿名社会网络有向图 G' 。为了减少信息损失, 本文基于 GraphX 图数据处理平台, 通过节点间的信息传递实现虚拟节点对的合并删除, 提高所发布数据的可用性。

图 7 匿名社会网络有向图 G' Fig. 7 Anonymous social network directed graph G'

信息传递的数据结构用五元组 (dstid, srcid, hops, community, tags) 表示, 称五元组为 n -跳邻居列表 (n -Hops Neighborhood Table, HNT), 其中, dstid 表示目的节点 ID, srcid 表示源节点 ID, hops 表示跳数, community 表示源节点所属社区, tags 表示节点标志位 (初始时 $tags = 0$), $tags = 1$ 表示源节点和目的节点均为虚拟节点。

初始化匿名图 G' 的结果如图 8 所示 (仅列出部分虚拟节点初始 HNT (n -Hops Neighborhood Table Entry))。在初始时, 节点的 dstid 和 srcid 都是节点的 ID, hops = 0, tags = 0, 如节点 f_2 的 HNT = $\{f_2, f_2, 0, 1, 0\}$ 。

图 8 有向图 G' 的初始化结果Fig. 8 Initialization result of directed graph G'

定义 6 (有向图层次社区熵) 用有向图层次社区熵 (DGHCE) 量化添加边操作对社区层次结构的影响, 记作 $DGHCE(G, H_e)$, 其计算公式为:

$$\text{DGHCE}(G, H_G) = - \sum_{i=1}^{|V|-1} \frac{|T_r^R| \cdot |T_r^L| \cdot 2P_r}{|E|} \lg \frac{|T_r^R| \cdot |T_r^L| \cdot 2P_r}{|E|} \quad (3)$$

定义 7 (有向图层次社区熵变化值) 用 DGHCE 的变化值 UL 衡量虚拟节点对合并删除后添加的边操作造成的信息损失, 其计算公式为:

$$\text{UL}(G, G') = \left| \text{DGHCE}(G, H_G) - \text{DGHCE}(G', H_{G'}) \right| \quad (4)$$

定义 8 (虚拟节点对合并删除条件 VNMD C) 设存在边 $\langle u, f_w \rangle$ 和边 $\langle f_x, v \rangle$, 虚拟节点对 (f_w, f_x) 能够进行合并删除, 当且仅当 $\forall (f_w, f_x) \in \text{VirtualSet}$ 满足以下 3 个条件:

- 1) $f_w, f_x \notin \text{VirtualRDD}$;
- 2) $\langle u, v \rangle \notin \text{EdgeRDD}$;
- 3) $u \neq v$ 。

定理 1 对于虚拟节点对 (f_w, f_x) , VNMD C 条件是 (f_w, f_x) 能够合并删除的充分条件。

证明 如图 9(a) 所示, 合并删除虚拟节点对 (f_w, f_x) , 需删除边 $\langle u, f_w \rangle$ 、 $\langle f_x, v \rangle$ 并添加边 $\langle u, v \rangle$, 但是有向图 G 中已经存在边 $\langle u, v \rangle$, 故 (f_w, f_x) 不能合并删除。同理, 如图 9(b) 所示, 虚拟节点 f_w, f_x 均与节点 u 相连, 不能添加边 $\langle u, v \rangle$, 故 (f_w, f_x) 不能合并删除。因此, 当且仅当虚拟节点对 (f_w, f_x) 满足 VNMD C 条件时, (f_w, f_x) 能够合并删除, 如图 9(c) 所示。

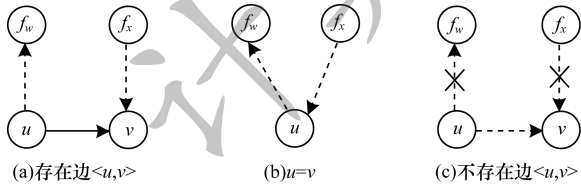


图 9 虚拟节点对之间的 3 种情况

Fig. 9 Three cases between virtual node pairs

通过节点间的信息传递得到每次迭代的虚拟节点对集合 VirtualSet, 判断 $\forall (f_w, f_x) \in \text{VirtualSet}$ 是否满足 VNMD C 条件, 将满足条件的虚拟节点对放入候选虚拟节点集合 CandidateSet 中。选择虚拟节点对合并删除算法 Select Merge_Delete(CandidateSet) 描述如下:

算法 3 Select Merge_Delete(CandidateSet) 算法

输入 候选合并删除节点集合 CandidateSet

输出 虚拟节点对 (f_w, f_x)

1. $M = \text{Number of same community in CandidateSet}$;
2. $N = \text{CandidateSet.size}$;
3. if ($N > 1$) then
4. if ($M > 1 \parallel M = 0$) then
5. $(f_w, f_x) = \text{the min(UL) from CandidateSet}$;
6. return (f_w, f_x) ;
7. end if
8. if ($M = 1$) then
9. return (f_w, f_x) ;

10. end if

11. else

12. return (f_w, f_x)

若候选虚拟节点对集合 CandidateSet 中虚拟节点对个数大于 1, 执行算法 3 第 4 行 ~ 第 7 行, 对于同一社区 (或均为不同社区) 的虚拟节点对计算 DGHCE 值, 选择 UL 值小的虚拟节点对进行合并删除。如果虚拟节点对个数为 1, 执行算法 3 的第 8 行 ~ 第 12 行直接选择虚拟节点对 (f_w, f_x) 。

3.4 KIODA 算法步骤

本文基于层次社区结构的大规模社会网络 K-出入度匿名算法 KIODA 描述如下:

算法 4 KIODA 算法

输入 原始图 G , 匿名参数 k

输出 匿名图 G^*

1. 基于 Construst HRG 算法生成 H_G ;
2. 基于 Sequence Partition(G, k) 算法生成 K-出入度匿名序列 d^* ;
3. 根据匿名序列添加虚拟节点, 得到匿名图 G' ;
4. 初始化匿名图 G' , $\text{CandidateSet} = \emptyset$, $\text{VirtualRDD} = \emptyset$;
5. for SuperStep = 1 to 6 do
6. $\text{Dst. Message} \leftarrow \text{Src. Message}$; // 源节点将更新的 // HNT E 信息发送到目的节点
7. for (Message from Dst. HNT E) do
8. if (Message.Tags == 1) then
9. $\text{Dst. VirtualSet} \leftarrow \text{Message}$;
10. end if
11. end for
12. for (Message from Dst. VirtualSet) do
13. $f_w = \text{Message.srcid}$, $f_x = \text{Message.disid}$;
14. if (f_w, f_x) satisfy VNMD C then
15. $\text{CandidateSet} \leftarrow (f_w, f_x)$;
16. end if
17. end for
18. if $\text{CandidateSet.size} > 0$ then
19. $(f_w, f_x) = \text{Select Merge_Delete(CandidateSet)}$;
20. $G'. \text{EdgeRDD. Remove } \langle u, f_w \rangle$;
21. $G'. \text{EdgeRDD. Remove } \langle f_x, v \rangle$;
22. $G'. \text{EdgeRDD. Add } \langle u, v \rangle$;
23. $\text{VirtualRDD. Add } (f_w, f_x)$;
24. $\text{VoteToHalt}(f_w, f_x)$;
25. end if
26. end for
27. return G^*

KIODA 算法具体步骤如下:

1) 在 Superstep = 0 时, 节点初始化得到初始边集合 EdgeRDD。

2) 在 Superstep = 1 时, 进行第 1 次迭代。节点收到自己的 1 跳邻居信息, 生成 1-跳邻居列表。第 1 次迭代标志位均为 0, $\text{VirtualSet} = \emptyset$, $\text{CandidateSet} = \emptyset$ 。

3) 在 Superstep = 2 时, 进行第 2 次迭代, 2-跳邻居列表如表 1 所示 (仅列出部分虚拟节点), 查看是否存在 tags = 1。迭代得到 $\text{VirtualSet} = \{(f_2, f_1)\}$, 由于虚

拟节点 f_2, f_1 均与节点 1 相连, 不满足 VNMD C 条件, 故不能进行合并删除, $\text{CandidateSet} = \emptyset$ 。

表 1 2-跳邻居列表
Table 1 2-hops neighbor list

节点	dstid	srcid	hops	community	tags
f_2	f_2	f_1	2	1	1
f_4	f_4	4	2	1	0
f_6	f_6	12	2	3	0
f_6	f_6	14	2	3	0
f_8	f_8	8	2	2	0
f_8	f_8	9	2	2	0
f_9	f_9	1	2	1	0
f_9	f_9	5	2	1	0
f_{10}	f_{10}	3	2	1	0
f_{10}	f_{10}	5	2	1	0
f_{10}	f_{10}	12	2	3	0

4) 在 Superstep = 3 时, 进行第 3 次迭代, 3-跳邻居列表如表 2 所示 (仅列出 tags = 1 的节点), 迭代得到 $\text{VirtualSet} = \{(f_{10}, f_{11}), (f_9, f_1), (f_6, f_5), (f_6, f_{11})\}$ 。

表 2 3-跳邻居列表
Table 2 3-hops neighbor list

节点	dstid	srcid	hops	community	tags
f_6	f_6	f_{11}	3	3	1
f_6	f_6	f_5	3	3	1
f_9	f_9	f_1	3	1	1
f_{10}	f_{10}	f_{11}	3	3	1

虚拟节点对 (f_{10}, f_{11}) 不满足 VNMD C 条件, 不能合并删除, 故 $\text{CandidateSet} = \{(f_9, f_1), (f_6, f_5), (f_6, f_{11})\}$ 。执行 KIODA 算法第 20 行 ~ 第 24 行合并删除虚拟节点对 (f_9, f_1) , 将边 $\langle 3, 1 \rangle$ 添加到 EdgeRDD 中, 删除边 $\langle f_1, 1 \rangle$ 、 $\langle 3, f_9 \rangle$; 虚拟节点 f_9, f_1 添加到 VirtualRDD 中, 状态设为 InActive, 停止迭代。虚拟节点对 (f_6, f_5) 、 (f_6, f_{11}) 执行算法 3 计算 DGHCE 的变化情况, 选择 UL 值小的虚拟节点对进行合并删除。若合并删除 (f_6, f_5) , 有向图层次社区树 H_G 的连接概率 P_1 由 $1/2$ 变为 1 ; 若合并删除 (f_6, f_{11}) , P_2 由 $1/2$ 变为 $3/4$, 如图 10 所示。

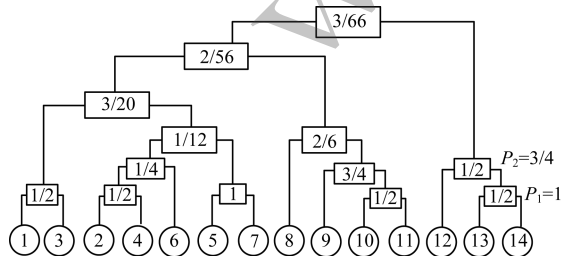


图 10 层次社区熵示意图
Fig. 10 Schematic diagram of hierarchical community entropy

经计算, 原始图的 $\text{DGHCE} = 3.574\ 15$, $\text{DGHCE}_{f_6, f_5} = 3.593\ 27$, $\text{UL}_{f_6, f_5} = 0.019\ 12$, $\text{DGHCE}_{f_6, f_{11}} = 3.563\ 07$, $\text{UL}_{f_6, f_{11}} = 0.011\ 08$, 因此, 选择合并删除虚拟节点对 (f_6, f_{11}) , 此时 $\text{VirtualRDD} = \{f_9, f_1, f_6, f_{11}\}$ 。第 3 次迭代停止, 结果如图 11 所示。

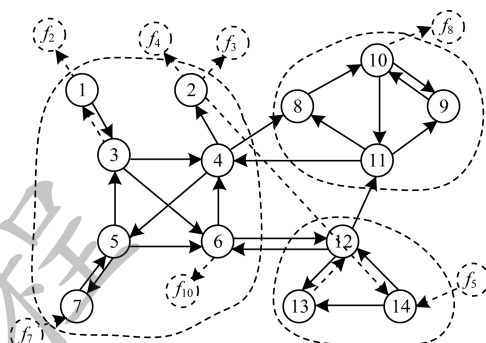


图 11 第 3 次迭代结果
Fig. 11 Results of the third iteration

KIODA 算法在迭代 6 次后停止虚拟节点对合并删除, 得到如图 12 所示的匿名社会网络有向图 G^* 。

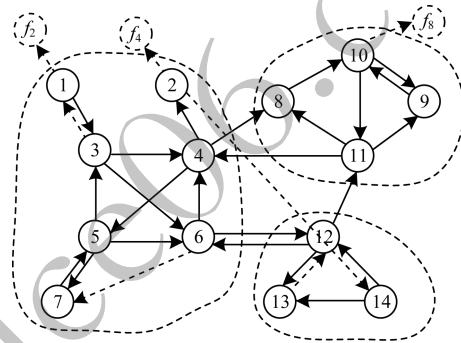


图 12 最终匿名社会网络有向图 G^*
Fig. 12 Eventually anonymous social network directed graph G^*

4 实验结果与分析

本节将 KIODA 算法与快速 K-度匿名 (KDA) 算法^[10]、VA (VertexAddition) 算法^[23] 进行性能分析与比较。

4.1 实验设置

本次实验采用斯坦福大学公开的 2 个真实社会网络有向图数据集 Eu-Email 和 Epinions。Eu-Email 网络是欧洲大型研究机构在 2003 年 10 月—2005 年 5 月期间的电子邮件数据, 其中, 给定一组电子邮件消息, 每个节点对应一个电子邮件地址, 有向边 $\langle i, j \rangle$ 表示节点 i 向 j 发送一条消息。Epinions 网络是一个消费者的在线评论社交网络, 网站成员可以决定是否“互相信任”, 有向边 $\langle u, v \rangle$ 表示用户 u 信任用户 v 。表 3 所示为数据集相关统计数据。实验使用的处理工具是 GraphX, 算法运行环境为 15 个计算节点, 1.8 GHz CPU, 16 GB RAM, Hadoop 2.7.2, Spark2.2.0, 采用 Scala 2.11.12 编程。

表 3 数据集相关统计数据
Table 3 Statistical data of datasets

参数	Epinions 数据集	Eu-Email 数据集
节点数目	75 879	265 214
边数目	508 837	420 045
最大入度数	3 035	7 631
最大出度数	1 801	930
平均出/入度数	6.71	1.58
(0,1)节点数目	18 328	170 768
(1,0)节点数目	11 774	36 922
平均聚类系数	0.137 8	0.067 1
三角形数目	1 624 481	267 313

4.2 算法性能分析

图 13 所示为随着隐私值 k 的变化, 算法的运行时间对比结果。从图 13 可以看出, 随着 k 值的增加, 各算法的运行时间也随之增加, 其中, KDA 算法运行时间最短, KIODA 算法运行时间最长。这是因为 KIODA 算法首先要基于层次社区结构对原始图进行社区划分, 然后再对出入度序列分组匿名, 最后合并删除虚拟节点对, 随着 k 值的增加, 需要添加更多的虚拟节点, 同时为了减少信息损失, 要对更多的虚拟节点对进行合并删除操作。因此, KIODA 算法的执行时间略大于 KDA 算法和 VA 算法, 但是总体而言, KIODA 算法的运行时间在可接受范围内。

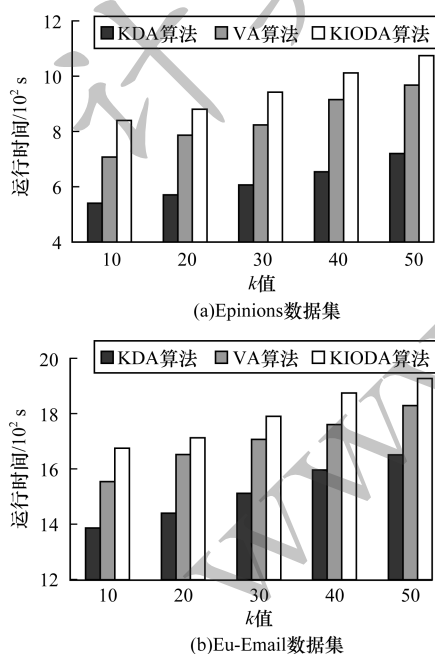


图 13 3 种算法的运行时间对比结果

Fig. 13 Running time comparison results of three algorithms

4.3 信息损失分析

为了比较算法在匿名过程中的信息损失, 测试算法匿名后平均出/入度数的变化情况。图 14 所示为随着 k 值的增加, 3 种算法在不同数据集的平均

出/入度数对比结果。从图 14 可以看出, 随着 k 值的增加, VA 算法在匿名后节点的平均度数变化较大, 而 KDA 算法和 KIODA 算法匿名后节点的平均度数更接近于原始平均度数。这是因为 VA 算法通过添加节点来实现 K-匿名, 匿名后造成节点的平均度数变化大, 对图结构的影响较大, 信息损失大; KDA 算法和 KIODA 算法尽可能减少图的修改, 因此, 匿名后平均节点度数变化较小, 图信息损失较小。

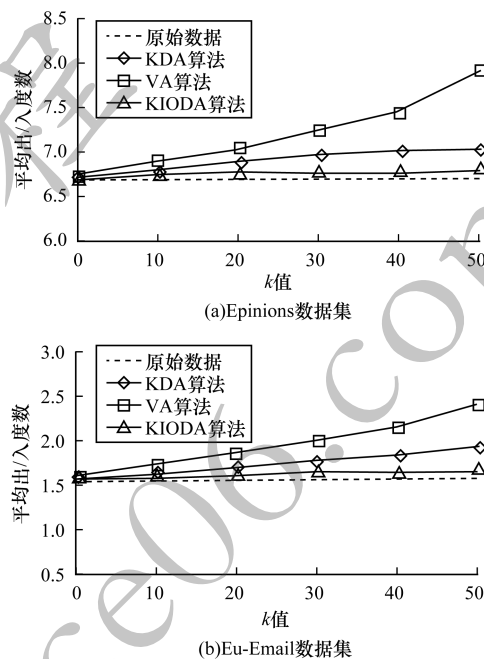


图 14 3 种算法的平均出/入度数变化情况

Fig. 14 Changes of average out/in degrees of three algorithms

聚集系数 (Clustering Coefficient, CC) 是用来描述一个图中顶点之间聚集程度的系数。匿名后平均聚集系数 (Average Clustering Coefficient, ACC) 变化越小, 匿名过程对社区变化的影响越小。为了衡量算法匿名过程中在图结构方面的信息损失, 测试匿名图 ACC 的变化率如下:

$$\text{Changeradio} = |\text{ACC}^* - \text{ACC}| / \text{ACC} \times 100\% \quad (5)$$

其中, ACC^* 表示匿名后的平均聚类系数值。

图 15 所示为 3 种算法匿名后 ACC 的变化率情况, 从图 15 可以看出, KDA 算法和 VA 算法在匿名后 ACC 的变化率均大于 KIODA 算法。这是因为 KDA 算法对图的修改最少, VA 算法添加节点实现最佳 K-匿名化。KDA 算法和 VA 算法在匿名时仅考虑满足 K-匿名, 而不考虑节点所属社区情况, 因此, ACC 变化率较大, 匿名后社区结构可用性低。随着 k 值的增加, KIODA 算法的 ACC 变化率很小, 因此, KIODA 算法更好地保证了有向图的图结构性性质, 减少了社区结构的信息损失。

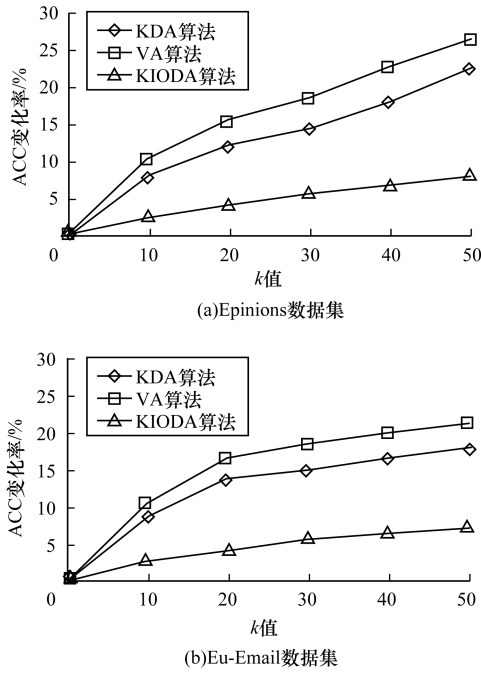


图 15 3 种算法的平均聚集系数变化率情况
Fig. 15 Change ratio of average clustering coefficient of three algorithms

4.4 数据可用性分析

$\mu_i (0 = \mu_1 \leq \mu_2 \leq \dots \leq \mu_m \leq m)$ 是拉普拉斯矩阵 L 的特征值, L 的第二小特征值 (μ_2) 是其重要的一个特征值, 用来表示社区的分离方式。图 16 所示为随着 k 值的增加, 3 种算法在不同数据集上的 μ_2 值与原始图的相似性比较结果。

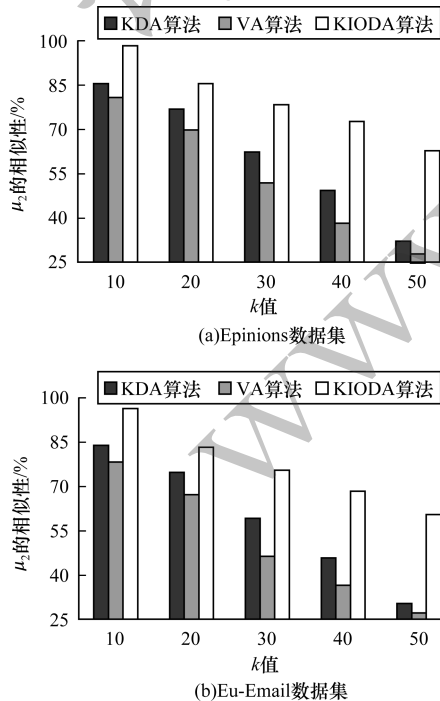


图 16 3 种算法的 μ_2 相似性比较结果
Fig. 16 μ_2 similarity comparison results of three algorithms

从图 16 可以看出, KIODA 算法的 μ_2 值与原始图更相似。这是因为 KIODA 算法在合并删除虚拟节点时考虑了原始图的社区结构, 而 KDA 算法和 VA 算法在匿名时忽略了对社区结构的保护, 给社区结构造成了很大的损失。

在匿名过程中, 为了衡量算法在社区结构方面的可用性, 引入精度指标 (Precision index)^[24] 衡量节点所属社区变化情况。若节点在原始图所属社区与匿名后的社区相同, 则 $\rho_{I_{nv}(v)=I_{pv}(v)}$ 值为 1; 若节点在原始图所属社区与匿名后的社区不同, 则 $\rho_{I_{nv}(v)=I_{pv}(v)}$ 的值为 0。精度指标的范围在 $[0, 1]$ 之间, 若精度指标的值为 0, 则原始图与匿名图的社区划分完全不同; 若精度指标的值为 1, 则原始图与匿名图的社区划分相同。精度指标计算公式如下:

$$\text{Precision index} = \frac{1}{n} \sum_{v \in G} \rho_{I_{nv}(v)=I_{pv}(v)} \quad (6)$$

图 17 所示为 3 种算法匿名后节点所属社区的变化情况。从图 17 可以看出, KDA 算法和 VA 算法在匿名过程中没有考虑节点的社区情况, 因此匿名后节点所属社区变化率很大。KIODA 算法在合并删除虚拟节点时尽可能保证了节点所属社区不变, 因此匿名后社区的精度指标更接近于 1, 其更好地保持了匿名图在社区划分方面的数据可用性。

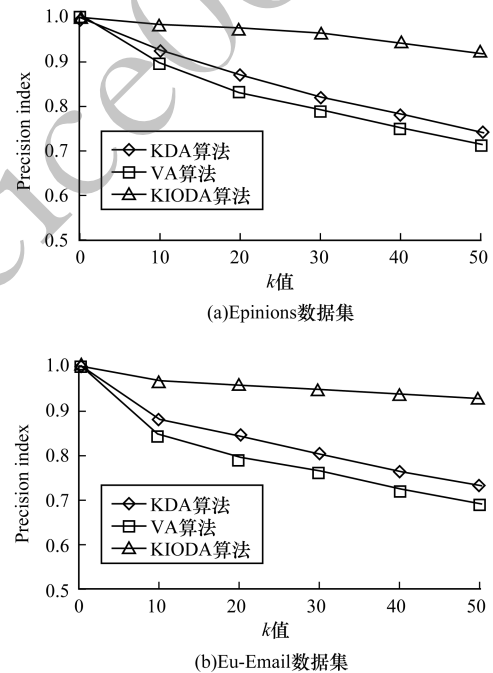


图 17 3 种算法的节点所属社区变化情况
Fig. 17 Change of community of nodes of three algorithms

5 结束语

针对大规模社会网络有向图的隐私保护问题, 本文构建一种新的出入度攻击模型并提出基于层次社区结构的大规模社会网络 K-出入度匿名算法。该算法基于层次社区结构划分社区, 根据贪心算法分组并

匿名出入度序列,对虚拟节点对进行合并删除以减少信息损失,在合并删除虚拟节点对的过程中保持原始图中节点所属社区不变。基于真实社会网络数据集的实验结果表明,在分布式并行环境下执行该算法,能够对大规模社会网络数据进行并行高效的隐私保护,与传统 K-度匿名算法相比,其提高了大规模社会网络有向图数据的处理效率,更好地保证了数据发布时社区结构分析结果的可用性,在节点平均出/入度数变化、平均聚类系数、拉普拉斯矩阵第二小特征值(μ_2)以及社区结构保护等方面都取得了很好的效果。本文算法在保证社区结构分析结果可用性的基础上保护了节点的隐私信息,但是,目前该算法仅针对具有相同隐私保护等级的用户,下一步将针对不同隐私保护等级的用户,研究个性化的 K-出入度匿名方法。

参考文献

- [1] BACKSTROM L, DWORK C, KLEINBERG J, et al. Wherefore art thou R3579X? anonymized social networks, hidden patterns, and structural steganography [J]. Communications of the ACM, 2011, 54(12): 133-141.
- [2] SALAS J, TORRA V. Graphic sequences, distances and k-degree anonymity [J]. Discrete Applied Mathematics, 2015, 188: 25-31.
- [3] BHATTACHARYA M, MANI P. Preserving privacy in social network graph with K-anonymize degree sequence generation [C]//Proceedings of 2015 International Conference on Software, Knowledge, Information Management and Applications. Washington D. C., USA: IEEE Press, 2015: 1-7.
- [4] MACWAN K R, PATEL S J. K-NMF anonymization in social network data publishing [J]. The Computer Journal, 2018, 61(4): 601-613.
- [5] YANG Dianhui. Large scale social network privacy protection based on heuristic analysis [D]. Xi'an: University of Electronic Science and Technology of China, 2013. (in Chinese)
杨典辉. 基于启发式分析的大规模社会网络隐私保护[D]. 西安: 电子科技大学, 2013.
- [6] ZHANG Xiaolin, ZHANG Wenchao, ZHANG Chen, et al. D-GSPerturb: a distributed social privacy protection algorithm based on graph structure perturbation [J]. Journal of Computers, 2017, 28(5): 51-61.
- [7] SUN Yongjiao, YUAN Ye, WANG Guoren, et al. Splitting anonymization: a novel privacy-preserving approach of social network [J]. Knowledge and Information Systems, 2016, 47(3): 595-623.
- [8] KASIVISWANATHAN S P, NISSIM K, RASKHODNIKOVA S, et al. Analyzing graphs with node differential privacy [M]. Berlin, Germany: Springer, 2013: 457-476.
- [9] LIU Xiangyu, WANG Bin, YANG Xiaochun. Survey on privacy preserving techniques for publishing social network data [J]. Journal of Software, 2014, 25(3): 576-590. (in Chinese)
刘向宇, 王斌, 杨晓春. 社会网络数据发布隐私保护技术综述[J]. 软件学报, 2014, 25(3): 576-590.
- [10] LIU K, TERZI E. Towards identity anonymization on graphs [C]//Proceedings of 2008 ACM SIGMOD International Conference on Management of Data. New York, USA: ACM Press, 2008: 93-106.
- [11] CASAS-ROMA J, HERRERA-JOANCOMARTÍ J, TORRA V. k-degree anonymity and edge selection: improving data utility in large networks [J]. Knowledge and Information Systems, 2017, 50(2): 447-474.
- [12] MACWAN K R, PATEL S J. K-degree anonymity model for social network data publishing [J]. Advances in Electrical and Computer Engineering, 2017, 17(4): 117-124.
- [13] YU D R, ZHAO H X, WANG L E, et al. A hierarchical k-anonymous technique of graphlet structural perception in social network publishing [M]. Berlin, Germany: Springer, 2019: 224-239.
- [14] GONG Weihua, LAN Xuefeng, PEI Xiaobing, et al. Privacy preservation method based on k-degree anonymity in social networks [J]. Acta Electronica Sinica, 2016, 44(6): 1437-1444. (in Chinese)
龚卫华, 兰雪峰, 裴小兵, 等. 基于 k-度匿名的社会网络隐私保护方法[J]. 电子学报, 2016, 44(6): 1437-1444.
- [15] LIU Xiangyu, LI Jiajia, AN Yunzhe, et al. Efficient algorithm on anonymizing social networks with reachability preservation [J]. Journal of Software, 2016, 27(8): 1904-1921. (in Chinese)
刘向宇, 李佳佳, 安云哲, 等. 一种保持结点可达性的高效社会网络图匿名算法[J]. 软件学报, 2016, 27(8): 1904-1921.
- [16] ZHANG Xiaolin, HE Xiaoyu, ZHANG Huanxiang, et al. PLRD-(k, m): distributed k-degree-m-label anonymity with protecting link relationships [J]. Journal of Frontiers of Computer Science and Technology, 2019, 13(1): 70-82. (in Chinese)
张晓琳, 何晓玉, 张焕香, 等. PLRD-(k, m): 保护链接关系的分布式 k-度-m-标签匿名方法[J]. 计算机科学与探索, 2019, 13(1): 70-82.
- [17] YING Xiaowei, WU Xintao. On link privacy in randomizing social networks [J]. Knowledge and Information Systems, 2011, 28(3): 645-663.
- [18] CAMPAN A, ALUFAISAN Y, TRUTA T M. Preserving communities in anonymized social networks [J]. Transactions on Data Privacy, 2015, 8(1): 55-87.
- [19] WANG Huanjie, LIU Peng, LIN Shan, et al. A local-perturbation anonymizing approach to preserving community structure in released social networks [M]. Berlin, Germany: Springer, 2017: 36-45.
- [20] KUMAR S, KUMAR P. Upper approximation based privacy preserving in online social networks [J]. Expert Systems with Applications, 2017, 88: 276-289.
- [21] ROUSSEAU F, CASAS-ROMA J, VAZIRGIANNIS M. Community-preserving anonymization of graphs [J]. Knowledge and Information Systems, 2018, 54(2): 315-343.
- [22] CLAUSET A, MOORE C, NEWMAN M E J. Hierarchical structure and the prediction of missing links in networks [J]. Nature, 2008, 453(7191): 98-101.
- [23] CHESTER S, KAPRON B M, RAMESH G, et al. Why Waldo befriended the dummy? k-anonymization of social networks with pseudo-nodes [J]. Social Network Analysis and Mining, 2013, 3(3): 381-399.
- [24] CAI Bingjing, WANG Haiying, ZHENG Huiru, et al. Evaluation repeated random walks in community detection of social networks [C]//Proceedings of 2010 International Conference on Machine Learning and Cybernetics. Washington D. C., USA: IEEE Press, 2010: 1849-1854.