软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2013,24(12):2883−2896 [doi: 10.3724/SP.J.1001.2013.04411]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

面向缺失数据的数据匿名方法*

龚奇源, 杨 明, 罗军舟

(东南大学 计算机科学与工程学院,江苏 南京 211118)

通讯作者: 龚奇源, E-mail: gongqiyuan@seu.edu.cn

摘 要: 在数据发布过程中,为了防止隐私泄露,需要对数据的准标识符属性进行匿名化,以降低链接攻击风险,实现对数据所有者敏感属性的匿名保护.现有数据匿名方法都建立在数据无缺失的假设基础上,在数据存在缺失的情况下会直接丢弃相关的记录,造成了匿名化前后数据特性不一致.针对缺失数据匿名方法进行研究,基于 *k*-匿名模型提出面向缺失数据的数据匿名方法 KAIM(*k*-anonymity for incomplete mircrodata),在保留包含缺失记录的前提下,使在同一属性上缺失的记录尽量被分配到同一分组参与泛化.该方法将分组泛化前后的信息熵变化作为距离,基于改进的 *k*-member 算法对数据进行聚类分组,最后通过基于泛化层次的局部泛化算法对组内数据进行泛化.实际数据集的大量实验结果表明,KAIM 造成信息缺损仅为现有算法的 43.8%,可以最大程度地保障匿名化前后数据特性不变.

关键词: 数据匿名;缺失数据;聚类;*k*-匿名

中图法分类号: TP309 **文献标识码**: A

中文引用格式: 龚奇源,杨明,罗军舟.面向缺失数据的数据匿名方法.软件学报,2013,24(12):2883−2896. http://www.jos.org.cn/1000-9825/4411.htm

英文引用格式: Gong QY, Yang M, Luo JZ. Data anonymization approach for incomplete microdata. Ruan Jian Xue Bao/Journal of Software, 2013,24(12):2883−2896 (in Chinese). http://www.jos.org.cn/1000-9825/4411.htm

Data Anonymization Approach for Incomplete Microdata

GONG Qi-Yuan, YANG Ming, LUO Jun-Zhou

(School of Computer Science and Engineering, Southeast University, Nanjing 211118, China)

Corresponding author: GONG Qi-Yuan, E-mail: gongqiyuan@seu.edu.cn

Abstract: To protect privacy against linking attacks, quasi-identifier attributes of microdata should be anonymized in privacy preserving data publishing. Although lots of algorithms have been proposed in this area, few of them can handle incomplete microdata. Most existing algorithms simply delete records with missing values, causing large information loss. This paper proposes a novel data anonymization approach called KAIM (*k*-anonymity for incomplete microdata), for incomplete microdata based on *k*-member algorithm and information entropy distance. Instead of deleting any records, KAIM effectively clusters records with similar characteristics together to minimize information loss, and then generalizes all records with local recording scheme. Results of extensive experiments base on real dataset show that KAIM causes only 43.8% information loss compared with previous algorithms for incomplete microdata, validating that KAIM performs much better than existing algorithms on the utility of anonymized dataset.

Key words: data anonymization; incomplete microdata; clustering; *k*-anonmity

数据采集和共享技术的快速发展,为各种组织机构间的合作和研究工作提供了巨大的便利,同时也增加了隐私信息泄露的风险.例如,医院会把收集的诊疗信息发布给医疗研究机构,供其进行疾病分析和预测方面的研

* 基金项目: 国家自然科学基金(61272054, 61202449, 61003257, 61320106007); 国家重点基础研究发展计划(973) (2010CB328104);国家高技术发展计划(863)(2013AA013503); 国家科技支撑计划(2010BAI88B03, 2011BAK21B02); 高等学校博士学科点专项科研基金(20110092130002); 江苏省网络与信息安全重点实验室(BM2003201); 教育部网络与信息集成重点实验室(93K-9)

收稿时间: 2012-02-21; 定稿时间: 2013-04-02

究,但是诊疗信息中可能涉及到用户的隐私信息.虽然在信息发布过程中,数据发布单位会消除个体标识符信息和某些敏感数据,但是通过多个公开数据集之间的链接攻击(linking attack)[1],还是会造成隐私信息泄露.文献[1]的研究表明,即使删除了标识符信息,攻击者仍然可以通过邮政编码、年龄和性别之类的准标识符属性与其他数据集进行联合,最终确定用户的敏感信息.为了防止这种攻击,数据匿名技术得到越来越多的关注,大量数据匿名模型和数据匿名算法也随之产生.

但是,现有数据匿名方法无法处理带有缺失的数据,在数据存在缺失的情况下会丢弃相关记录,而数据缺失在数据处理过程中是普遍存在的.采集用户数据时,如果用户拒绝提供某项具体数据,该属性上的数值就变为缺失.数据在传输和存储过程中,也会产生缺失.UCI 公用数据集[2]中与健康相关的 21 个常用数据集中,有 11 个数据集带有缺失值,部分数据集的缺失度超过 15%.直接删除含有缺失的记录会造成过度的数据缺损,例如,表 1 中的记录 1 和记录 2 都在 Age 属性上存在缺失,现有数据匿名方法会在预处理阶段删除这类数据,从而导致记录 1 和记录 2 的其他信息丢失.而事实上,我们可以将记录 1 和记录 2 放到同一个等价类中进行匿名化,保留大部分信息.

Table 1 Patient dataset

表 1 病人诊疗记录

Id	Age	Gender	Zipcode	Disease
1	*	M	12 000	Gastric ulcer
2	*	M	14 000	Dyspepsia
3	26	F	18 000	Pneumonia
4	28	M	19 000	Bronchitis
5	32	M	*	*
6	39	M	24 000	Pneumonia
7	41	*	*	Flu
8	36	F	22 000	Gastritis
9	48	F	*	Pneumonia
10	*	F	21 000	Flu

本文针对缺失数据匿名方法进行研究,基于 *k*-匿名模型[1]提出面向缺失数据的数据匿名方法 KAIM.在保留包含缺失记录的前提下,为避免缺失记录造成过高的泛化缺损,使在同一属性上缺失的记录尽量被分配到同一分组参与泛化.该方法将分组泛化前后的信息熵变化作为距离,基于改进的 *k*-member[3]算法对数据进行聚类分组,最后通过基于泛化层次的局部泛化算法对组内数据进行泛化.实际数据集的大量实验表明,KAIM 可以避免记录丢弃,同时,最大程度地保障匿名化后数据可用性.例如,表 1 中的数据经过现有数据匿名算法处理,最终得到的结果见表 2,大部分记录被丢弃;而通过 KAIM 算法的处理,见表 3,数据中记录数目保持不变,且大量可用信息得以保留.

Table 2 Released dataset

表 2 匿名发布的诊疗记录

Age	Gender	Zipcode	Disease
[20,30)	Person	[15000,20000)	Pneumonia
[20,30)	Person	[15000,20000)	Bronchitis
[30,40)	Person	[20000,25000)	Pneumonia
[30,40)	Person	[20000,25000)	Bronchitis

本文第 1 节介绍相关工作.第 2 节介绍数据缺失和数据匿名基本概念.第 3 节介绍并分析缺失数据的处理方法和 KAIM 算法.第 4 节分析和评价实验结果.第 5 节总结全文.

1 相关工作

数据发布中的隐私保护已经得到很多研究工作者的关注.Sweeney 和 Samarati[1]首先指出链接攻击带来的隐私泄露问题,并提出 *k*-匿名模型——通过保证每条记录都有至少 *k*–1 条记录与它相似,来确保发布数据受到链接攻击时不会泄露隐私信息.文献[4]给出了基于泛化空间完全搜索的 *k*-匿名算法 MinGen.Meyerson 等人在

文献[5]中证明了当 $k\geqslant 2$ 时,通过最小的泛化实现数据 k-匿名化的问题是 NP 难的.文献[6]给出了近似比为 $O(\log k)$的近似算法.文献[7]讨论了 k-匿名算法实现过程中 k 选取问题.文献[8]对 k-匿名中准标识符属性选取进行了分析和实现.文献[9,10]分别讨论了高维 k-匿名问题和多约束 k-匿名化问题.文献[11]给出了基于全局泛化空间剪枝的 k-匿名算法 Incognito.

实现 k-匿名的主要方法是数据泛化,针对这个问题,文献[4,11]给出了基于全局泛化的 k-匿名算法.全局泛化方法的搜索空间较小,产生的匿名化数据格式统一便于分析,但是会造成较高的数据缺损.鉴于全局泛化造成的数据缺损过高,文献[12]提出了局部泛化技术,用于降低泛化开销.文献[13]在之前泛化算法的基础上提出了多维度泛化技术,通过多维度划分将泛化粒度再次缩小.为了降低泛化带来的信息缺损问题,文献[14]提出了基于有损连接的数据匿名方法 Anatomy,文献[15]提出了基于同样思路且效果更好的匿名化方法 ANGEL.

在部分情况下,k-匿名无法确保隐私信息的安全.例如,当大部分记录具有相同的敏感属性取值时,攻击者能够以较高的概率推断出用户的隐私信息.因此,文献[16]在 k-匿名的基础上提出了(α,k)-匿名,保证发布数据在满足 k-匿名的同时,每个等价类中与任意敏感属性值相关的记录百分比不超过α.文献[17]提出了安全性更高的 l-diversity,保证每个等价类中敏感属性的不同取值至少有 l 个.文献[18]在 l-diversity 的基础上考虑敏感属性的分布问题,并提出了 t-Closeness,保证不同等价类中的敏感属性分布尽量接近于全局分布.文献[19]指出在数据增量发布的过程中,现有数据匿名模型会造成隐私泄露,并提出了 m-Invariance 型.文献[20]提出了个性化隐私保护(personalized privacy preservation)的匿名化模型.

以上提出的数据匿名技术主要针对完整数据集,对于含有缺失的不完整数据集,特别是缺失度很高的数据集,现有数据匿名技术会删除包含缺失的记录,造成大量数据丢弃.文献[21]虽然提出了支持缺失数据的处理方法,但是只局限于位图数据,且会造成较严重的数据缺损.本文假设 k-匿名能够满足安全发布需求,首次详细讨论了缺失数据的数据匿名问题,提出了面向缺失数据的数据匿名算法 KAIM.

2 基本定义

本文主要侧重于关系数据集.根据属性的特性,所有属性被分为 4 类:

(1) 标识个体身份的标识符属性(identifier),如身份证号码、姓名等,这类信息必须从发布数据集中移除;

(2) 联合起来能够唯一标识个体信息的属性,如 Age,Gender,Zipcode 等,称为准标识符 QI(quasi-identifier),需要进行匿名化处理;

(3) 包含个体隐私信息的属性 SA(sensitive attribute),如 Disease、家庭住址的等,在无法关联到个体时不会泄露隐私;

(4) 其他属性,不属于上述 3 类的属性,这类属性不需要做特殊处理.

为了方便,论文将需要发布的数据集标记为 T,用 T^*代表匿名化之后的数据,n 表示 T 中记录数目,m 标记 T 中 QI 数目,t 表示 T 中记录,v 表示 t 中某个具体取值.

2.1 数据缺失

不同领域对缺失度(missing rate)的定义有很大差异.为了避免混淆,本文采用数值缺失度和记录缺失度来衡量数据集中缺失的数值比例和含有缺失数值的记录比例,用*来表示缺失数据.

定义 1(数值缺失度(value missing rate,简称 VMR)). 设在准标识符属性组合 $QI_1,\ldots,QI_m$ 上,含有缺失数值的单元数目为 n_v,则我们定义数据集 T 的数值缺失度为 $VMR=n_v/mn$.

数据集的数据值缺失度可以衡量数据集中缺失数值在所有数值中占的比例,能够反映一个数据集的缺失程度.因为每个准标识符上的缺失情况不一,故数值缺失度受所选准标识符组合影响.

定义 2(记录缺失度(record missing rate,简称 RMR)). 设 T 中含有缺失数值的记录数目为 n_r,则我们定义数据集 T 的记录缺失度为 $RMR=n_r/n$.

记录缺失度可以用来衡量含有缺失的记录在数据集中所占比例.

一条记录中可能含有多个缺失数值,即 $n_rm\leqslant n_v\leqslant nm$;$VMR$ 和 RMR 之间的关系为 $0\leqslant RMR/VMR\leqslant 1$.

2.2 数据匿名

定义 3(等价类(equivalence class,简称 EC))[13]. 在准标识符属性组合 $QI_1,\ldots,QI_m$ 上,具有相同取值的记录组合在一起形成等价类.

数据表 T 由复数 EC_i 组成,EC_i 相当于 T 的子表.以$|EC_i|$表示等价类 EC_i 的大小,则$|EC_i|$越大,等价类受链接攻击的影响越低,组内数据越安全.对于某个孤立记录 t,没有其他记录在准标识符上与它具有相同取值,则 t 形成一个大小为 1 的等价类.

定义 4(k-匿名(k-anonymity))[1]. 如果数据表 T 中的任意记录 t,都能找到至少 k–1 条记录与 t 在准标识符上无法区分,则称该数据表 T 满足 k-匿名.

定理 1. 如果 T 中所有的等价类均满足 k-匿名,则 T 必然满足 k-匿名.

证明:由定义 3 可知,等价类中的记录在准标识符上无法区分.设 T 中的任意记录为 t,如果 $t\in EC_i$ 则至少能找到$|EC_i|$–1 条记录与 t 在准标识符上无法区分.又因为任意 EC_i 满足 k-匿名,则 EC_i 中无法区分的记录数大于等于 k,即$|EC_i|\geqslant k$.联合可得,T 中任意记录 t 至少能找到 k–1 条记录在准标识符上与其无法区分. □

在匿名化之前,T 中等价类大部分都不满足 k-匿名.为了达到数据匿名化的目的,需要通过数据匿名技术对原始数据进行转化,以达到数据匿名模型要求.现有数据匿名技术中,泛化技术最能反映原始数据集的特性,适用范围最广.

定义 5(泛化(generalization))[1]. 在数据匿名化过程中,用模糊的、范围的值取代原有数值的过程称为泛化.

泛化是数据挖掘领域中的概念,即用模糊数值取代确定值[1,3,6].对应泛化层次(generalization hierarchy,又称分类树)的结构而言,就是用高层节点替代低层节点.用越上层的节点来进行泛化,泛化程度越高,数据缺损越严重.泛化所需泛化层次如图 1 所示,文中假设准标识符泛化层次已经建立完毕.

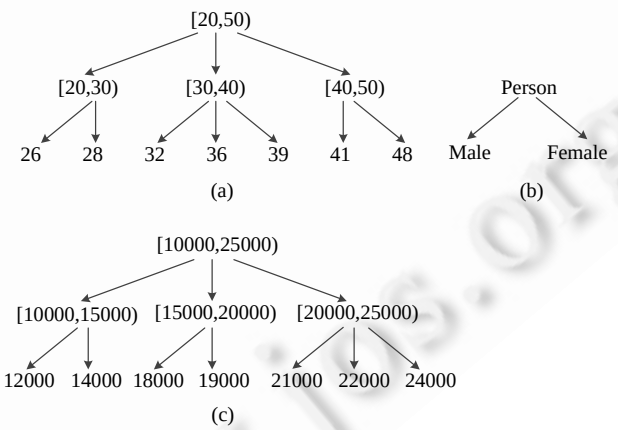


Fig.1 Generalization hierarchy

图 1 泛化层次结构

目前,泛化主要分为全局泛化和局部泛化:全局泛化技术实现难度较低,对泛化层次的遍历空间也比较小,但是会带来过度的泛化损失;局部泛化实现难度比全局泛化大,且遍历空间也比全局泛化大,但是灵活性更高,造成的数据缺损更低.考虑到数据可用性最大化和后期数据分析的难度,我们采用区域不重叠的局部泛化技术.

2.3 数据可用性

数据泛化会让等价类合并为更大的等价类,最终达到 k-匿名,但是也会造成数据可用性降低.目前,数据可用性可以用信息缺损度程度公式来度量,缺损度公式的定义如下:

定义 6(数据集信息缺损度(information loss,简称 ILoss))[22]. 设原始数据集为 T,经过泛化变为 T^*,则将 T 泛化为 T^*的所产生的信息缺损为 $ILoss(T^*)$:

$$ILoss(T^*)=\sum_{t^*\in T^*}\sum_{v^*\in t^*}ILoss(v^*) \tag{1}$$

$$ILoss(v^*)=\frac{cover(v^*)}{cover(QI)} \tag{2}$$

$ILoss(v^*)$表示单个数值的数据缺损;$cover(v^*)$表示 v^*涵盖的不同取值数量;而 $cover(QI)$表示该准标识符中不同取值的数量,每个 QI 的 $cover(QI)$为常数.且 $0\leqslant cover(v^*)\leqslant cover(QI)$,$0<cover(QI)\leqslant n$,因此,$0\leqslant ILoss(v^*)\leqslant 1$.

鉴于数值属性和离散属性的差异,可将公式(2)划分为公式(3)和公式(4):

- 数值属性信息缺损度:

$$ILoss(v^*)=\begin{cases}(v^*_{\max}-v^*_{\min}+1)/(QI_{\max}-QI_{\min}+1), & v\neq v^*\\ 0, & v=v^*\end{cases} \tag{3}$$

  $v^*_{\max}$ 和 $v^*_{\min}$ 分别表示 v^*代表区域的最小值和最大值,而 $QI_{\max}$ 和 $QI_{\min}$ 表示该属性上的最大值和最小值.根据公式(3),表 2 中的记录 1 在 Age 属性上的缺损度为 0.48.

- 离散属性信息缺损度:

$$ILoss(v^*)=\begin{cases}\Lambda(v^*)/\Lambda(QI), & v\neq v^*\\ 0, & v=v^*\end{cases} \tag{4}$$

  $\Lambda(v^*)$表示该节点在泛化层次中包含的叶子节点数目,$\Lambda(QI)$表示该属性根节点包含的叶子节点数目.根据公式(4),表 2 中的记录 1 在 gender 的信息缺损度是 1.

缺损度公式只提供缺损度的一个衡量,并没有给出缺损度相对于整个数据集的比例.例如,某数据集的缺损度是 120,该数值对于数据使用者没有参考价值.所以在这里,我们提出缺损比例公式.

定义 7(数据集信息缺损比例(information loss rate,简称 ILossRate)). 设原始数据集为 T,经过泛化变为 T^*,则将 T 泛化为 T^*的所产生的信息缺损相对于全数据集的比例为

$$ILossRate(T^*)=\frac{ILoss(T^*)}{ILoss(ALL)} \tag{5}$$

$ILoss(ALL)$代表丢失所有信息所带来的缺损度.

通过公式(1)~公式(4),可得 $ILoss(ALL)=nm$ 以及 $0\leqslant ILoss(T^*)\leqslant nm$,从而得到 $0\leqslant ILossRate(T^*)\leqslant 1$.

现有数据匿名算法在计算缺损度的过程中,会把删除记录造成的缺损度忽略.这个做法本身与数据缺损度衡量相悖,另外也会误导数据使用者.例如,常用的测试数据集美国公民收入数据原先有 48 842 条记录,滤去含有缺失数据的记录之后,剩下 45 522 个记录,大部分数据匿名算法没有把丢失 3 320 条记录造成的缺损计算到信息缺损中.当数据记录缺失度很严重时,现有数据匿名算法会删除大量的记录,造成数据可用性降低,但是最终输出的数据集缺损度却可能很低.为了解决这个问题,我们使用缺损度公式来衡量记录的丢弃.根据公式(2)~公式(4),当记录被删除时,认为记录为全部缺损,损失全部信息的记录在每个属性上的缺损度为 1,故整条记录的缺损度为 m.对应到缺损比例中,分母中有因子 m,故记录缺失带来的缺损度比例为 $1/n$.

为了便于区分,文中将缺损度划分为两类:抑制缺损度和泛化缺损度.抑制缺损度主要指删除造成的缺损度损失;泛化缺损度表示数据集在泛化过程中造成的数据缺损.

这里,我们用 $ILoss(G)$代表泛化缺损度,用 $ILoss(RecordLoss)$代表抑制缺损度:

$$ILoss(T^*)=ILoss(G)+ILoss(RecordLoss) \tag{6}$$

根据记录缺失度 RMR 的定义,丢失记录数为 n_r,则 $ILoss(RecordLoss)=n_r\times m$,对应到 $ILossRate(T^*)$中的结果为

$$ILossRate(T^*)=\frac{ILoss(G)}{nm}+\frac{n_r}{n} \tag{7}$$

在实验分析模块,为了完成算法性能对比,将 $ILoss(RecordLoss)$加入到缺损度计算中.

3 面向缺失数据的数据匿名方法

3.1 支持缺失数据的泛化层次

现有数据匿名算法默认的泛化层次对缺失数据*没有提供支持,如图 1 所示.为了让层次泛化技术能够处理缺失数据,需要明确*在泛化层次中的位置.参照文献[1,4]中对抑制(suppression)的定义,我们得到以下定理:

定理 2. *可以涵盖某个属性泛化层次的根节点.

证明:记根节点为 *root*,*代表缺失值,即任意取值,取值为整个有效值域,故*和泛化层次最高层的语义相同.为了证明*可以涵盖泛化层次根节点,我们可以假设根节点中存在某个取值 $a\in(root-*)$,且 a 在该属性的有效值域内.由于*具有任意取值的特性,故 $a\in*$,假设与实际矛盾,故 *root* 中的所有取值都在*内. □

根据定理 2,可以建立如图 2 所示泛化层次,在这个层次内,*和层次根节点具有相同的高度.通过这样的泛化层次,我们就可以对含有缺失数据的数据集进行匿名化操作.

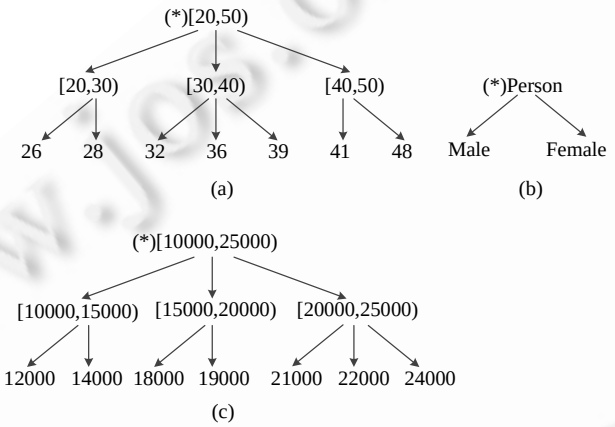


Fig.2 Generalization hierarchy for incomplete microdata

图 2 支持缺失数据的泛化层次结构

3.2 支持缺失数据的信息熵距离

本文所用聚类算法是基于划分的聚类,聚类算法中的距离定义会直接影响聚类效果.虽然记录距离问题已经有多种计算方法[24−26],但是大部分方法都不支持缺失数据,如果直接应用到缺失数据中会遇到很大的问题.例如在欧式距离中,对于缺失数值没有办法定义,如果按照字符串内容不相同来进行对比,会忽略语义特性,即*与 Private 的距离和*到 Fedora-Work 的距离一样.

为了解决上述问题,我们将信息熵引入作为距离,用以衡量记录之间泛化距离.信息熵作为距离可以很好地回避缺失数据在距离计算的时候带来的影响,保留数据的语义特性,也可以回避高维诅咒问题.

定义 8(单元信息熵). 设 QI_j 中的数值 v 被泛化为 v^*,p_l 表示 v^*落在(泛化层次)叶子节点 l 的概率,则 v^*的信息熵为

$$Info(v^*)=\left|\sum_{l}^{cover(v^*)} p_l\times\log(p_l)\right| \tag{8}$$

$cover(v^*)$表示 v^*包含的叶子节点数目,$cover(v^*)$由泛化层次结构确定.p_l 可以从 T 的统计数据中获取,例如 Gender 属性上,male 的概率是 0.6685;对于缺失数据在统计中占有的比例,我们假设缺失数据的归属不可确定,故在统计过程中忽略不计.根据公式(8),我们可以得知叶子节点的信息熵为 0.

定理 3. 单元信息熵随着数据泛化程度的增加,单调递增.

证明:通过对公式(2)~公式(4)的分析可以得知,随着泛化程度增加,v^*包含的叶子节增加,$cover(v^*)$单调递增.对于固定数据集 T,$|p_l\times\log(p_l)|$是常数,所以 $Info(v^*)$随着 v^*泛化程度的增加,单调递增. □

定理 4. 信息熵作为距离可以处理缺失数据问题.

证明:根据定理 2,缺失数据*的信息不确定性最大,即信息熵最大.公式(8)中,缺失数据具有最大的 $cover(v^*)$,所以 $Info$ 最大,与定理 2 的结果相符.当信息经过泛化之后,由于缺失数据处于泛化层次的最高层,泛化不会影响缺失数据,故信息熵不变,还是处于最大值. □

定义 9(单元泛化度). 设 QI_j 中的数值 v 被泛化为 v^*,则 v 被泛化为 v^*的泛化度为

$$GenRate(v,v^*)=\frac{Info(v^*)}{Info(v)+c} \tag{9}$$

v^*包含的叶子节点越多,$GenRate(v,v^*)$越大.考虑到叶子节点 $Info(v)$为 0,论文加入变量 $c\geqslant 0$.通过信息熵的比值,我们可以抵消不同属性在信息熵上的基数.变量 c 的取值需要考虑属性差异,论文中我们取 $c=|p_v\log p_v|$,p_v为叶子节点概率.

定义 10(记录泛化度). 设 T 中记录 t 被泛化为 t^*,则 t 被泛化为 t^*的泛化度为

$$GenRate(t,t^*)=\sum_{j}^{m} w_j\times GenRate(t[QI_j],t^*[QI_j]) \tag{10}$$

$t[QI_j]$,$t^*[QI_j]$分别表示 t 和 t^*在 QI_j 上面的取值,w_j 表示 QI_j 的权重.泛化度 $GenRate(t,t^*)$可以衡量记录 t 泛化为 t^*损失信息的比例.在权重位置的情况下,我们默认 w_j 为 1.

定义 11(记录间距离). 设有记录 t_1 和 t_2,如果用 t^*表示 t_1 和 t_2 泛化后形成等价类的结果,则 t_1 和 t_2 之间的距离为

$$D(t_1,t_2)=GenRate(t_1,t^*)+GenRate(t_2,t^*) \tag{11}$$

我们定义 t_1 和 t_2 间距离为将这两个记录泛化至相同 QI 之后泛化度损失之和.

定义 12(分组中心记录). 设 g_i 代表 G_i 的中心记录,则 g_i 表示将 G_i 转换为等价类 EC_i 之后,任意记录准标识符的取值.

分组中心记录是一个虚拟存在的记录,当泛化完成时,等价类中任意记录的准标识符取值和中心记录相同.中心记录可以很好地代表分组现在的泛化状态,同时又减少全部泛化带来的计算开销.中心记录的敏感属性没有实际意义.

定义 13(记录到分组的距离). 设 g_i 为分组 G_i 中心记录,如果记录 $t\notin G_i$,则 t 到 G_i 的距离可以定义为

$$D(t,G_i)=D(t,t^*)+|G_i|\times D(g_i,t^*) \tag{12}$$

其中,$|G_i|$代表 G_i 中记录的数目,这里的 t^*是 g_i 和 t 的泛化结果.$D(t,G_i)$的距离等于 $D(t,t^*)$与$|G_i|$倍的 $D(g_i,t^*)$之和.因为泛化之后的 G_i 分组中所有记录具有相同的 QI 取值,故 G_i 中任意记录到 t^*的距离都是 $D(g_i,t^*)$.

根据上述定义,我们可以根据 T 中的统计数据确定 p_l,计算记录间的距离和记录到分组的距离,从而决定记录的归属.在将记录划分到分组中之后,也不需要立刻进行组内泛化操作,只要更新中心记录就可以完成分组状态更新.

3.3 KAIM算法

本节中,我们会给出一种面向缺失数据的数据匿名方法,其基于聚类思想对记录进行最优化分组.之后,通过局部泛化算法对各分组进行泛化.在聚类模块中,我们通过基于信息熵距离的聚类完成对数据的分簇,保证簇内信息熵距离最小、簇外信息熵距离最大.之后,通过基于泛化层次的局部泛化算法对各分组进行泛化,使得每个分组内部在准标识符上具有相同取值,从而形成等价类.表 1 的数据经过 KAIM 算法匿名化后可得表 3,数据特性被最大限度保留.

KAIM 算法经过修改之后,可以满足于 l-diversity 和 m-Invariance.由于篇幅有限,这里不作介绍.

Table 3 KAIM released dataset

表 3 KAIM 发布数据

Age	Gender	Zipcode	Disease
*	M	[10000,15000)	Gastric ulcer
*	M	[10000,15000)	Dyspepsia
[20,30)	*	[15000,20000)	Pneumonia
[20,30)	*	[15000,20000)	Bronchitis
[30,40)	M	*	*
[30,40)	M	*	Pneumonia
*	F	[20000,25000)	Gastritis
*	F	[20000,25000)	Flu
[40,50)	*	*	Flu
[40,50)	*	*	Pneumonia

3.3.1 聚类算法

KAIM 算法的聚类算法主要基于修改后的 *k*-member 算法,该聚类算法是一个面向 *k*-匿名问题的基于划分的聚类算法.我们以第 3.2 节中提出的信息熵作为距离,用 *k*-member 算法对记录进行分组.通过聚类算法的处理,原始的发布数据被划分为大小至少为 *k* 的 *n*/*k* 个分组.这些分组遵循两个特性:

(1) 组内距离最小;

(2) 组间距离最大化.

聚类过程如图 3 所示:

- 首先,从 T 中随机选取记录 t_i 建立候选分组 G_i,将 t_i 从 T 中移除;
- 然后,从 T 中选取到 G_i 距离最小的记录 r;
- 将 r 加入到 G_i,并更新 G_i 的中心记录;
- 将 r 从 T 中移除;
- 如果 G_i 中记录数目超过 k,则将 G_i 加入到 G 中.

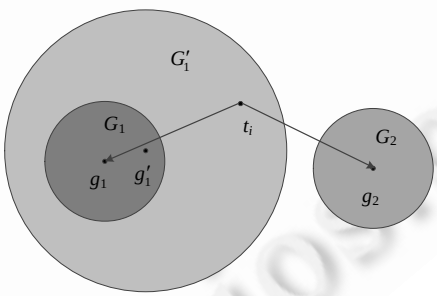


Fig.3 Clustering starge

图 3 聚类阶段

算法 1 的步骤 2 流程需要重复 K 次.至算法 1 的步骤 3 结束,K 个分组已经建立完毕,且每个分组中都包含 k 个记录;而 T 中可能还有部分记录没有分类.算法 1 的步骤 4、步骤 5 主要完成对 T 中剩余记录的处理,将这些记录加入到与他们距离最小的分组中,并每次加入都需要更新中心记录.更新中心记录就是寻找分组内最低共同父节点的过程,这个过程的复杂度为 $O(km)$.更新中心记录的目的有 2 个:(1) 方便计算记录与分组的距离;(2) 泛化阶段不需要重新计算泛化目标.聚类算法执行完毕时,T 中没有剩余记录,G 中包含 K 个分组且每个分组都包含至少 k 个记录.

算法 1. 聚类算法.

输入:k 和 T;

输出:$G\{G_1,\ldots,G_K\}$,$\forall G_i \in G, |G_i| \geq k$,且 $\sum_i^K |G_i| = |G|$.

1. $K=|T|/k$;
2. **for** i=0 to K
   从 T 中随机选取一个记录 t_i;
   建立 G_i,使 $G_i=\{t_i\}$,$T=T-\{t_i\}$;
   while $|G_i|<k$
   $r=T$ 中 $D(r,G_i)$最小的记录;
   $G_i=G_i+\{r\}$,$T=T-\{r\}$;
   更新 G_i 的中心记录 g_i;
   end while
   $G=G+\{G_i\}$;
3. **end for**
4. **while** $|T|\geqslant 0$
   从 T 中随机选取一个记录 r;
   $bestG=G$ 中 $D(r,G_i)$最小的分组;
   $bestG=bestG+\{r\}$,$T=T-\{r\}$;
   更新 $bestG$ 中心记录;
5. **end while**

3.3.2 泛化算法

泛化层次指导下的泛化需要用父亲节点的数值取代叶子节点的数值.为了使得组内的记录在相同准标识符属性上具有相同取值并且保留数据可用性,需要找到叶子节点的共同父节点.分组泛化过程中的中心记录计算,也就是寻找最低共同父节点的过程.在泛化算法中,需要对分组内记录的准标识符取值进行泛化,而泛化的计算在聚类算法执行过程中就已经完成(中心记录更新).故只需要将分组中心记录赋值给分组内所有记录即可.算法 2 的步骤 2 执行完毕之后,分组内的每个记录在准标识符上都具有相同的取值.

算法 2. 泛化算法.

输入:k 和 $G\{G_1,\ldots,G_K\}$;

输出:$T^*\{EC_1,\ldots,EC_K\}$.

1. **for** i = 0 to K
   //假设 g_i 已经计算好
   for each $t\in G_i$
   $t=g_i$;
   end for
2. **end for**

3.4 算法分析

3.4.1 正确性分析

聚类算法最终输出 n/k 个分组 $G\{G_1,\ldots,G_K\}$,且每个分组都不小于 k.在泛化阶段,分组内部准标识符都被泛化为相同值,每个分组都形成一个等价类,即 $T^*\{EC_1,\ldots,EC_K\}$.由于泛化过程中记录数目不变,则可以得知,$EC_1,\ldots,EC_K$ 都不小于 k,故根据定理 1 所有等价类满足 k-匿名定义,整个数据集达到 k-匿名要求,即通过 KAIM 算法匿名化的数据能够安全发布.

3.4.2 复杂度分析

设原始数据集 T 中的记录数目为$|T|=n$,准标识符维数为$|QI|=m$,聚类算法执行完步骤 3 之后,得到 K 个分组,时间复杂度为 $O(kKn)$.算法在步骤 4、步骤 5 内,需要扫描$(n\%k)$次,每次都要计算与各分组的距离,并重新计算中心记录,执行复杂度为 $O(k^2m)$.聚类算法的复杂度为 $O(kKn)+O(k^2m)$,代入 $K=n/k$,得 $O(n^2)+O(k^2m)$,因为 $n>>k$,

$n>>m$,故聚类算法的复杂度为 $O(n^2)$.由于中心记录已经计算,泛化算法只需要进行全分组遍历赋值就行,时间复杂度为 $O(n)$.因此,KAIM 算法的总体时间复杂度为 $O(n^2)+O(n)=O(n^2)$,算法复杂度为 $O(n^2)$.

4 实验结果

本节通过实验分析 KAIM 的性能,将与文献[3]提出的 *k*-member 算法进行比较.根据第 3.4.1 节中的分析可知,KAIM 和 *k*-member 都满足 *k*-匿名,即两个算法具有相同的匿名度,因此,我们主要从数据可用性和缺失数据影响来进行比较.为分析 KAIM 算法中随机因子的影响,论文增加了稳定性分析.*k*-member 算法是基于聚类的局部泛化算法,该算法在缺损度和复杂度方面的表现都很好.

实验所使用的数据集为 UCI 机器学习数据库中的 Adult 数据集.该数据集属于 1996 年美国人口统计数据的一部分,在数据匿名研究中被广泛使用.Adult 数据集本身包含 15 个属性,实验中,我们保留 9 个属性:Age, Gender,Race,Marital Status,Education,Native Country,Work Class,Occupation,Salary Class.其中,Age 属性为有序属性,其他属性均为无序属性.实验中,分别取 m=2,…,8 个属性作为准标识符,Salary Class 作为敏感属性,实验数据集中各个属性的特性见表 4.该数据集的数值缺失度 *VMR* 为 1.65%,记录缺失度 *RMR* 为 7.41%.在后面的实验中,为了测试 KAIM 在高缺失度情况下的性能,我们在原数据集的基础上随机加入定量的缺失.实验的硬件环境为 Intel Pentium IV 2.8GHz CPU,2G RAM,操作系统为 Microsoft Windows XP.实验中的算法 KAIM 和 *k*-member 均由 Java 实现.

Table 4 Attributes selected for experiment

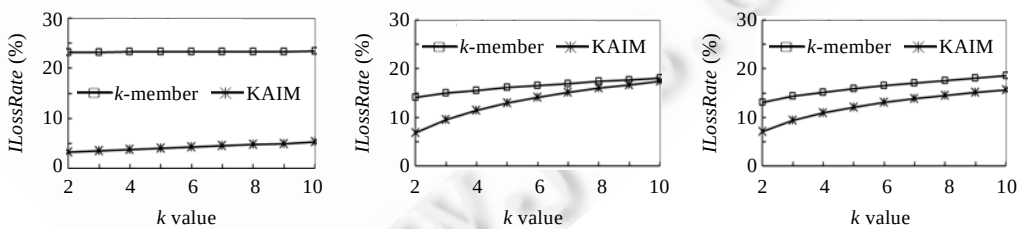
表 4 实验所选属性

属性	Age	Work class	Educaton	Marital status	Occupation	Race	Gender	Native country	Salary class
基数	90	8	16	7	15	5	2	41	2

4.1 数据可用性分析

在数据匿名化过程中,*k* 的取值、准标识符维度和数据集大小都会影响数据匿名化效果.在此,我们分析 KAIM 和 *k*-member 算法造成的数据缺损比例在这 3 个变量上的趋势.

图 4(a)~图 4(c)给出了|*T*|=48842,当|*QI*|=2,|*QI*|=5 和|*QI*|=8 时,*k* 值的变化对 KAIM 算法和 *k*-member 算法的信息缺损影响.图中信息缺损用缺损度比例 *ILossRate* 来表示,其中,*ILossRate* 的含义见定义 7.



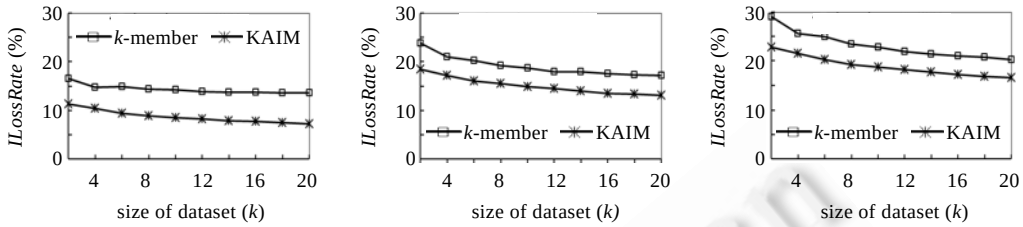
(a) |*T*|=48842, |*QI*|=2 (b) |*T*|=48842, |*QI*|=5 (c) |*T*|=48842, |*QI*|=8

Fig.4 Information loss when varying the value of *k*

图 4 *k* 值变化下缺损度的变化

我们可以看到,在同等条件下,KAIM 算法造成的缺损均比 *k*-member 算法小,即说明 KAIM 在数据可用性方面优于 *k*-member 算法.另外,随着 *k* 的增加,KAIM 和 *k*-member 算法的信息缺损均随之增加.这是由于当 *k* 的值增加时,每个等价类中包含的记录数目增加,泛化带来的损失也随之增加.

图 5(a)~图 5(c)给出了|*QI*|=8,当 *k*=2,*k*=5 和 *k*=10 时,数据集大小变化对 KAIM 算法和 *k*-member 算法的信息缺损影响.测试数据为从 Adult 数据集中随机抽取的 10 个大小不同的数据集.



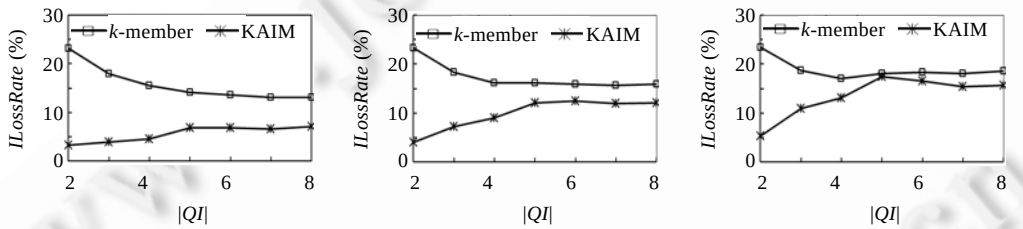
(a) k =2, $|QI|$=8 (b) k =5, $|QI|$=8 (c) k =10, $|QI|$=8

Fig.5 Information loss when varying the size of dataset

图 5 数据集变化下缺损度的变化

我们可以看出,随着数据集大小的增加,KAIM 和 *k*-member 算法造成的数据缺损比例缓慢降低.这是因为聚类效果随着数据集的增加而趋向于稳定,并最终影响匿名化结果.

图 6(a)~图 6(c)给出了$|T|$=48842,当 k=2,k=5 和 k=10 时,$|QI|$变化对 KAIM 算法和 *k*-member 算法的信息缺损影响.



(a) k =2, $|T|$=48842 (b) k =5, $|T|$=48842 (c) k =10, $|T|$=48842

Fig.6 Information loss when varying the size of $|QI|$

图 6 准标识符维数变化对缺损度的影响

随着准标识符数量的增加,KAIM 算法的造成的缺损度都会上升,而 *k*-member 算法造成的缺损度随着准标识符数量的增加有下降的趋势.这是因为 *k*-member 算法从缺损度角度考虑分组,准标识符维度增加会让划分更趋向于缺损度低的方式;而 KAIM 从信息熵比例的角度去分组,维度增加只会增加泛化缺损.

4.2 缺失度影响

图 7(a)给出了$|T|$=48842,当 k=5,$|QI|$=8 时,数据集数值缺失度 VMR 的增加对 *k*-member 算法缺损度中泛化缺损和抑制缺损的比例影响.

可以看出,随着 VMR 的增加,基于完美数据的 *k*-member 算法会因为删除了不完整记录而造成大量的数据缺损,导致最终参与匿名化的数据减少,从而在泛化缺损度上,*k*-member 算法呈现下降趋势,而抑制缺损度和总体数据缺损呈急剧上升趋势.

从这里可以看出,如果不把删除记录造成的缺损计算在内,即使数据已经严重缺损,我们得到的缺损度都很低,这会对数据使用者产生误导,最终影响数据分析结果的价值.

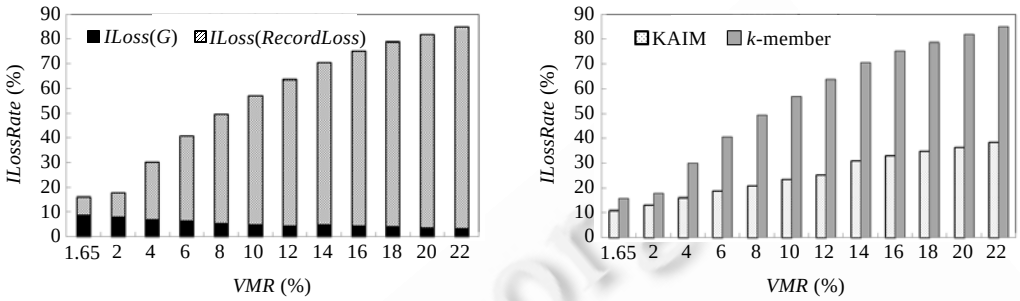
图 7(b)给出了$|T|$=48842,当 k=5,$|QI|$=8 时,当数据集 VMR 增加时,KAIM 和 *k*-member 算法的缺损度的变化.

我们可以看出:

- KAIM 造成的缺损比例上升得较缓慢,可以很稳定地完成数据匿名工作;
- 而基于完美数据的 *k*-member 会因为删除含有缺失的记录,造成缺损度急剧增加.

通过缺损度对比我们发现:

- KAIM 造成的缺损度约为 *k*-member 的 43.8%,该数值随着数据集缺失度的增加而减少;
- 当缺失度进一步增加时,*k*-member 将损失所有信息,输出空记录;而 KAIM 造成的缺损度虽然会增加,但是总体缺损度还在可以容忍的范围内.



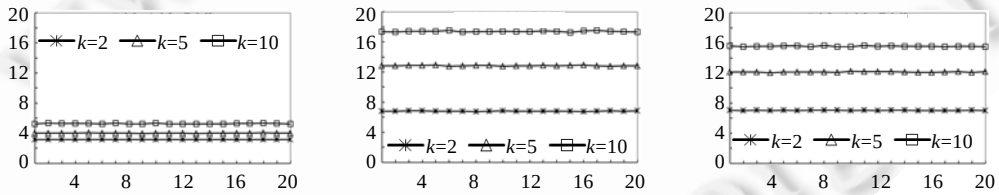
(a) k =5, |*QI*|=8 (b) k =5, |*QI*|=8

Fig.7 Information loss when varying VMR

图 7 缺失度变化对缺损度的影响

4.3 稳定性分析

KAIM 算法在聚类阶段两次用到了随机变量(建立分组和剩余元组处理).随机变量的引入,会对最终结果的稳定性造成影响.图 8(a)~图 8(c)给出了|*T*|=48842,当 k=2,k=5 和 k=10 时,在不同|*QI*|上,重复 20 次执行 KAIM 算法得到的结果.



(a) |*QI*|=2, |*T*|=48842 (b) |*QI*|=5, |*T*|=48842 (c) |*QI*|=10, |*T*|=48842

Fig.8 Effect of randomization on KAIM

图 8 随机变量对 KAIM 的影响

我们可以看出,20 次实验的结果相对稳定,KAIM 中使用随机变量对结果稳定性的影响较小.

5 结束语

针对缺损数据,本文提出了一种基于聚类的数据匿名方法 KAIM.该方法可以在兼容缺失数据的前提下,保证发布数据集满足 *k*-匿名模型,能够很好地满足数据发布过程中的隐私保护需求,避免链接攻击造成的隐私泄露.实际数据集的大量实验证明,KAIM 算法在缺失数据集上比现有数据匿名算法更加有效.在下一步工作中,我们会提高 KAIM 算法在高维准标识符条件下的计算效率,解决该算法在 k 值较高情况下的缺损度升高问题.

References:

[1] Sweeney L. *k*-Anonymity: A model for protecting privacy. Int'l Journal on Uncertain, Fuzziness and Knowledge-Based Systems, 2002,10(5):557−570. [doi: 10.1142/S0218488502001648]

[2] University of California irvine machine learning repository. http://archive.ics.uci.edu/ml/

[3] Byun JW, Kamra A, Bertino E, Li NH. Efficient *k*-anonymization using clustering techniques. In: Proc. of the 12th Int'l Conf. on Database Systems For Advanced Applications (DASFAA). Springer-Verlag, 2007. 188−200. [10.1007/978-3-540-71703-4_18]

[4] Sweeney L. Achieving *k*-anonymity privacy protection using generalization and suppression. Int'l Journal on Uncertain, Fuzziness Knowledge-based Systems, 2002,10(5):571−588. [doi: 10.1142/S021848850200165X]

[5] Meyerson A, Williams R. On the complexity of optimal K-anonymity. In: Proc. of the ACM Sigmod-Sigact-Sigart Symp. on Principles of Database Systems (PODS). ACM Press, 2004. 223−228. [doi: 10.1145/1055558.1055591]

[6] Park H, Shim K. Approximate algorithms for *k*-anonymity. In: Proc. of the ACM SIGMOD Int'l Conf. on Management of Data (SIGMOD). ACM Press, 2007. 67−78. [doi: 10.1145/1247480.1247490]

[7] Dewri R, Ray I, Ray I, Whitley D. On the optimal selection of *k* in the *k*-anonymity problem. In: Proc. of the IEEE 24th Int'l Conf. on Data Engineering (ICDE). IEEE, 2008. 1364−1366. [doi: 10.1109/ICDE.2008.4497557]

[8] Motwani R, Xu Y. Efficient algorithms for masking and finding quasi-identifiers. In: Proc. of the Conf. on Very Large Data Bases (VLDB). VLDB Endowment, 2007.

[9] Aggarwal CC. On *k*-anonymity and the curse of dimensionality. In: Proc. of the 31st Int'l Conf. on Very Large Data Bases (VLDB). VLDB Endowment, 2005. 901−909.

[10] Yang XC, Liu XY, Wang B, Yu G. *K*-Anonymization approaches for supporting multiple constraints. Ruan Jian Xue Bao/Journal of Software, 2006,17(5):1222−1231 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/17/1222.htm [doi: 10.1360/jos171222]

[11] LeFevre K, DeWitt DJ, Ramakrishnan R. Incognito: Efficient full-domain *K*-anonymity. In: Proc. of the ACM SIGMOD Int'l Conf. on Management of Data (SIGMOD). ACM Press, 2005. 49−60.

[12] Xu J, Wang W, Pei J, Wang XY, Shi BL, Fu AWC. Utility-Based anonymization using local recoding. In: Proc. of the 12th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining (SIGKDD). ACM Press, 2006. 785−790. [doi: 10.1145/1150402.1150504]

[13] LeFevre K, DeWitt DJ, Ramakrishnan R. Mondrian multidimensional *K*-anonymity. In: Proc. of the 22nd Int'l Conf. on Data Engineering (ICDE). IEEE, 2006. 25. [doi: 10.1109/ICDE.2006.101]

[14] Xiao XK, Tao Y. Anatomy: Simple and effective privacy preservation. In: Proc. of the 32nd Int'l Conf. on Very Large Data Bases (VLDB). VLDB Endowment, 2006. 139−150.

[15] Tao YF, Chen HK, Xiao X, Zhou S, Zhang D. ANGEL: Enhancing the utility of generalization for privacy preserving publication. IEEE Trans. on Knowledge and Data Engineering (TKDE), 2009,21(7):1073−1087. [doi: 10.1109/TKDE.2009.65]

[16] Wong RCW, Li JY, Fu AWC, Wang K. (α,*k*)-Anonymity: An enhanced *k*-anonymity model for privacy preserving data publishing. In: Proc. of the 12th ACM SIGKDD Int'l Conf. on Knowledge discovery and Data Mining (SIGKDD). ACM Press, 2006. 754−759. [doi: 10.1145/1150402.1150499]

[17] Machanavajjhala A, Kifer D, Gehrke J, Venkitasubramaniam M. *l*-Diversity: Privacy beyond *k*-anonymity. ACM Trans. on Knowledge Discovery Data (TKDD), 2007,1:3. [doi: 10.1145/1217299.1217302]

[18] Li NH, Li TC, Venkatasubramanian S. *t*-Closeness: Privacy beyond *k*-anonymity and *l*-diversity. In: Proc. of the IEEE 23rd Int'l Conf. on Data Engineering. IEEE, 2007. 106 −115. [doi: 10.1109/ICDE.2007.367856]

[19] Xiao XK, Tao YF. *m*-Invariance: Towards privacy preserving re-publication of dynamic datasets. In: Proc. of the ACM SIGMOD Int'l Conf. on Management of Data (SIGMOD). ACM Press, 2007. 689−700. [doi: 10.1145/1247480.1247556]

[20] Xiao XK, Tao YF. Personalized privacy preservation. In: Proc. of the ACM SIGMOD Int'l Conf. on Management of Data (SIGMOD). ACM Press, 2006. 229−240. [doi: 10.1145/1142473.1142500]

[21] Nergiz ME, Clifton C, Nergiz AE. Multirelational *k*-anonymity. IEEE Trans. on Knowledge and Data Engineering, 2009,21(8):1104−1117. [doi: 10.1109/TKDE.2008.210]

[22] Nergiz ME, Clifton C. Thoughts on *k*-anonymization. Data & Knowledge Engineering, 2007,63:622−645. [doi: 10.1016/j.datak.2007.03.009]

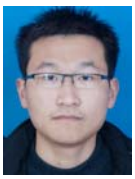
[23] Iyengar VS. Transforming data to satisfy privacy constraints. In: Proc. of the 8th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining (SIGKDD). ACM Press, 2002. 279−288. [doi: 10.1145/775047.775089]

[24] Aggarwal G, Panigrahy R, Feder T, Thomas D, Kenthapadi K, Khuller S, Zhu A. Achieving anonymity via clustering. ACM Trans. on Algorithms, 2010,6:1−19. [doi: 10.1145/1798596.1798602]

[25] Lin JL, Wei MC. An efficient clustering method for *k*-anonymization. In: Proc. of the Int'l Workshop on Privacy and Anonymity in Information Society. ACM Press, 2008. 46−50. [doi: 10.1145/1379287.1379297]

附中文参考文献:

[10] 杨晓春,刘向宇,王斌,于戈.支持多约束的 *K*-匿名化方法.软件学报,2006,17(5):1222−1231. http://www.jos.org.cn/1000-9825/17/1222.htm [doi: 10.1360/jos171222]



龚奇源(1986－),男,江苏江阴人,博士生,主要研究领域为数据匿名.
E-mail: gongqiyuan@seu.edu.cn

罗军舟(1960－),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为下一代网络体系结构,网络安全与管理,云计算,无线局域网.
E-mail: jluo@seu.edu.cn

杨明(1979－),男,博士,副教授,CCF 会员,主要研究领域为网络安全,无线网络.
E-mail: yangming2002@seu.edu.cn