ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.21, No.4, April 2010, pp.680−693
doi: 10.3724/SP.J.1001.2010.03508

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

一种基于聚类的数据匿名方法*

王智慧+, 许 俭, 汪 卫, 施伯乐

(复旦大学 计算机科学技术学院,上海 200433)

Clustering-Based Approach for Data Anonymization

WANG Zhi-Hui+, XU Jian, WANG Wei, SHI Bai-Le

(School of Computer Science, Fudan University, Shanghai 200433, China)

+ Corresponding author: E-mail: zhhwang@fudan.edu.cn

Wang ZH, Xu J, Wang W, Shi BL. Clustering-Based approach for data anonymization. *Journal of Software*, 2010,21(4):680−693. http://www.jos.org.cn/1000-9825/3508.htm

Abstract: To prevent the disclosure of privacy, it requires preserving the anonymity of sensitive attributes in data sharing. The attribute values on quasi-identifiers often have to be generalized before data sharing to avoid linking attack, and thus to achieve the anonymity in data sharing. Data generalization increases the uncertainty of attribute values, and results in the loss of information to some extent. Traditional data generalization is often based on the predefined hierarchy, which causes over-generalization and too much unnecessary information loss. In this paper, the attributes in a quasi-identifier are classified into two categories, ordered attributes and unordered attributes. More flexible strategies for data generalization are proposed for them, respectively. At the same time, the loss of information is defined quantitatively based on the change of uncertainty of attribute values during data generalization. Furthermore, data anonymization is modeled by a clustering problem with special constraints. A clustering-based approach, called *L*-clustering, is presented for the *l*-diversity model. *L*-clustering can meet the requirement of preserving anonymity of sensitive attributes in data sharing, and reduce greatly the amount of information loss resulting from data generalization for implementing data anonymization.

Key words: data anonymization; quasi-identifier; linking attack; clustering; information loss

摘 要: 为了防止个人隐私的泄漏,在数据共享前需要对其在准标识符上的属性值作数据概化处理,以消除链接攻击,实现在共享中对敏感属性的匿名保护.概化处理增加了属性值的不确定性,不可避免地会造成一定的信息损失.传统的数据概化处理大都建立在预先定义的概念层次结构的基础上,会造成过度概化,带来许多不必要的信息损失.将准标识符中的属性分为有序属性和无序属性两种类型,分别给出了更为灵活的相应数据概化策略.同时,通过考察数据概化前后属性值不确定性程度的变化,量化地定义了数据概化带来的信息损失.在此基础上,将数据匿名问

* Supported by the National Natural Science Foundation of China under Grant No.60673133 (国家自然科学基金); the National Basic Research Program of China under Grant No.2005CB321905 (国家重点基础研究发展计划(973)); the Specialized Research Fund for the Doctoral Program of Higher Education of China under Grant No.200802461146 (高等学校博士学科点专项科研基金); the Shanghai Rising-Star Program of China under Grant No.05QMX1405 (上海市青年科技启明星计划); the Shanghai Leading Academic Discipline Project of China under Grant No.B114 (上海市重点学科建设项目)

Received 2007-06-21; Accepted 2008-10-08

题转化为带特定约束的聚类问题.针对 *l*-多样模型,提出了一种基于聚类的数据匿名方法 *L*-clustering.该方法能够满足在数据共享中对敏感属性的匿名保护需求,同时能够很好地降低实现匿名保护时概化处理所带来的信息损失.

关键词: 数据匿名;准标识符;链接攻击;聚类;信息损失

中图法分类号: TP311 **文献标识码:** A

数据电子化和网络技术的发展,使得数据共享较之以前更为容易.但如果数据中含有涉及个体隐私的敏感属性(如医疗记录数据中的疾病诊断信息),共享这些数据会导致个人隐私的泄漏.因此,需要在共享之前对数据加以处理,对其中的敏感属性实现匿名保护.

如果仅仅将原始数据中能够唯一标识个体的属性(也称为标识符,如姓名、身份证号码等)去除,并不能有效地实现匿名保护.文献[1,2]指出,在数据集中常常存在一些称为准标识符(quasi-identifier)的非敏感属性的组合,通过准标识符,可以在数据集中确定与个体相对应的数据记录.因此,攻击者如果已知数据集中某个体在准标识符上的属性值,就可能推出该个体的敏感属性值,从而造成个人隐私泄漏,这种情况被称为链接攻击(linking attack)[1,2].例如,属性组(出生日期、性别、居住地邮编)就可以构成一个准标识符.研究[2]表明,约 87%的美国居民通过该准标识符能够被唯一确定.为了消除链接攻击,Sweenty 等人提出了 *k*-匿名模型(*k*-anonymity)[2].随后,Machanavajjhala 等人发现 *k*-匿名模型仍然存在隐私泄漏,于是提出了 *l*-多样模型(*l*-diversity)[3].

实现匿名保护通常要对数据在准标识符上的属性值作概化处理(generalization),即用较为抽象概括的属性值来代替原本具体的属性值.例如,用区间值[20~30]来代替 22,25,28 等具体年龄值.文献[4]中提出的消除处理(suppression)可看作是一种特殊的概化处理,即用空值(或者特殊符号“*”)来代替原始数据值.经过概化处理后,使得多个数据记录在准标识符上具有相同的属性值.这样可以阻止链接攻击,达到对敏感属性匿名保护的目的.但是,过度的概化会造成不必要的信息损失,降低共享数据的可用性.

针对数据共享中敏感属性的匿名保护,本文提出了一种基于聚类的数据匿名方法.该方法能够很好地满足 *l*-多样模型的匿名保护要求,防止与个体相关的敏感属性值的泄漏.同时,该方法消除了传统概化处理时的概念层次结构限制,采取更为灵活的数据概化策略,并基于聚类的思想来寻找合适的概化方案.实验结果表明,与现有满足 *l*-多样模型的数据匿名方法相比,该方法在实现匿名保护时能够有效地减少概化处理所带来的信息损失.

本文第 1 节介绍相关工作.第 2 节介绍与数据匿名保护相关的基本概念.第 3 节针对有序属性和无序属性,分别给出了相应的概化策略,并量化地定义了数据概化带来的信息损失.第 4 节针对 *l*-多样模型,提出了一种基于聚类的数据匿名方法.第 5 节分析和评价实验结果.第 6 节总结全文.

1 相关工作

近年来,数据共享中敏感属性的匿名保护已经得到了许多研究工作者的关注.文献[5,6]的研究表明,最优数据匿名问题(即在实现对敏感属性匿名保护的同时,使得信息损失最小化)是 NP 难问题.围绕如何降低匿名保护时的信息损失,已有多种启发式数据匿名方法被提了出来,具体有文献[7]提出的基于遗传算法的方法、文献[8,9]提出的自底向上的方法、文献[10]提出的自顶向下的方法、文献[11]提出的基于幂集空间搜索的方法、文献[12]提出的基于划分的方法、文献[5,13,14]提出的基于聚类的方法、文献[15]提出的针对多约束条件的方法、文献[16]提出的针对数据效用的方法等等.但是,这些方法都是基于 *k*-匿名模型,因此仍然存在隐私泄漏.

文献[3]给出了基于 Incognito[9]的 *l*-多样模型的实现方法.但是,Incognito 采取传统的基于概念层次结构的数据概化策略,在实现对敏感属性匿名保护时会造成许多不必要的信息损失.而本文的方法可以有效地处理共享数据,使其满足 *l*-多样模型,同时尽可能地减少信息损失.

文献[17]提出基于数据分解的匿名方法,其主要思想是,在共享时将原始数据集垂直地分割为两个部分,使得一部分只包含非敏感属性,另一部分只包含敏感属性,两者之间通过 Group-ID 关联.该方法可以降低信息损失,以满足面向聚集查询的应用需求.本文提出的方法可以与文献[17]的方法结合使用,即先用本文的方法对数

据集作聚类处理,随后不作真正的概化处理,而用文献[17]的方法来分解数据集.这样可以同时满足 l-多样模型的匿名保护要求以及面向聚集查询的应用需求.

2 数据匿名的基本概念

为了便于描述,我们以关系数据库为例,考虑待共享的数据集为一个多属性二维表,记做 D.设 Z 为 D 上的一个属性组,t 为 D 中一个元组,则 $t[Z]$表示 t 在 Z 上的属性值.D 中每个元组表示与某一个体相关的数据对象,且 D 中存在涉及个体隐私的敏感属性.例如,表 1 是表示病人诊断记录的数据集,其中 Disease 为敏感属性.

Table 1 Hospital records

表 1 病人诊断记录

	Age	ZipCode	Disease
t_1	51	12562	Heart disease
t_2	50	12552	Cancer
t_3	51	12532	Heart disease
t_4	54	12555	Cancer
t_5	54	12555	Heart disease
t_6	54	12555	Tracheitis
t_7	55	12532	Cancer
t_8	52	12561	Tracheitis
t_9	52	12533	Tracheitis
t_{10}	53	12553	Tracheitis

定义 1(准标识符). 一个准标识符是由 D 上若干个属性构成的属性组,记做 QI.准标识符中所包含的属性个数称为准标识符的维数,记做$|QI|$.

通过准标识符与其他渠道获得的信息进行连接,可以形成链接攻击.攻击者据此可推理出与个体相关的敏感属性值,从而导致隐私泄漏.例如,表 1 中的属性组(Age,ZipCode)可形成一个准标识符.当攻击者通过其他途径得知表中某病人的 Age 属性值为 51,ZipCode 属性值为 12562 时,从表中就可推出该病人患有 Heart Disease.准标识符的确定需要一定的领域知识,通常由数据拥有者和领域专家在对共享数据作匿名保护之前事先确定.本文的研究假设在 D 中的准标识符已经事先确定.同时为了简便描述起见,我们考虑 D 上有一个敏感属性,标记为 S.

定义 2(等价群及其代表元). 数据集 D 上的一个等价群为 D 中若干个元组的集合,其中每个元组在准标识符上具有相同的属性值.设 EQ 为 D 上的一个等价群,且 EQ 中所有元组在准标识符上具有相同属性值 A_{QI},则该等价群的代表元可以表述为这样的元组 t_g,其中,$t_g[QI]=A_{QI}$,$t_g[S]=Null$.

需要说明的是,等价群的代表元只是一个概念意义上的虚拟元组,并不要求其在数据集中出现.以表 1 为例,$\{t_5,t_6\}$可构成一个等价群 EQ,该等价群的代表元为 t_g=(54,12555,*Null*).即 t_g 在属性 Age,ZipCode,Disease 上的取值分别为 54,12555,*Null*.

定义 3(最大等价群). 设在数据集 D 上有等价群 EQ,其代表元为 t_g.等价群 EQ 为 D 上的一个最大等价群,当且仅当在 D 上不存在等价群 EQ_1,使得 $EQ \subset EQ_1$ 并且 $t_g=t_{g1}$(其中,t_{g1} 为 EQ_1 的代表元).

以表 1 为例,其中$\{t_5,t_6\}$可构成一个等价群,但不是最大等价群.因为$\{t_5,t_6\} \subset \{t_4,t_5,t_6\}$,并且两者的代表元均为(54,12555,*Null*).而$\{t_4,t_5,t_6\}$构成一个最大等价群,因为表中不存在一个等价群 EQ_1,使得$\{t_4,t_5,t_6\} \subset EQ_1$ 并且 t_{g1}=(54,12555,*Null*).

定义 4(k-匿名模型). 如果一个等价群包含有至少 k 个元组,则称该等价群是 k-匿名的;如果一个数据集中的每个最大等价群都是 k-匿名的,则称该数据集满足 k-匿名模型.

因为 k-匿名模型要求对于数据集中的每个元组都至少有 k–1 个其他元组与其准标识符上的属性值完全相同,即使利用链接攻击,也无法将数据集内的元组唯一地与某一个体相关联.因此,共享满足 k-匿名模型的数据,可以降低链接攻击所带来的隐私泄漏风险.但是,这样并不能完全防止隐私泄漏.例如,表 2 是表 1 的一个满足 3-匿名模型(即 k=3)的处理结果,其中,$\{t_1,t_2,t_3\}$,$\{t_4,t_5,t_6,t_7\}$和$\{t_8,t_9,t_{10}\}$分别构成 3 个最大等价群,且每个等价群中都

至少有 3 个元组.但是,如果从其他渠道获知表中某病人的 Age 和 ZipCode 属性值分别为 52 和 12533,则通过链接攻击,从表 2 中仍然可以唯一地确定该病人的 Disease 属性值为 Tracheitis.

针对 *k*-匿名模型的不足,文献[3]提出 *l*-多样模型.其实质是要求每个最大等价群中的元组除了在准标识符上的属性值相同以外,同时在敏感属性上至少有 *l* 个不同的取值.具体可以定义如下:

定义 5(*l*-多样模型). 如果一个等价群中的元组在敏感属性 *S* 上至少有 *l* 个不同的取值,则称该等价群是 *l*-多样的;如果一个数据集中的每个最大等价群都是 *l*-多样的,则称该数据集满足 *l*-多样模型.

文献[3]给出了几种实例来阐述 *l*-多样模型的思想.本文这里的定义取其基本思想,要求每个最大等价群中至少有 *l* 个不同的敏感属性值.例如,表 3 是表 1 的一个满足 3-多样模型(即 *l*=3)的处理结果,其中,(t_1,t_2,t_8),(t_3,t_7,t_9)和(t_4,t_5,t_6,t_{10})分别构成 3 个最大等价群.每个等价群中的元组在(Age,ZipCode)上具有相同的属性值,同时在属性 Disease 上具有 3 个不同值.显然,共享表 3 中的数据能够防止链接攻击所导致的隐私泄漏.

Table 2 Result meeting 3-anonymity

表 2 满足 3-匿名模型的结果

	Age	ZipCode	Disease
t_1	[50~51]	125**	Heart disease
t_2	[50~51]	125**	Cancer
t_3	[50~51]	125**	Heart disease
t_4	[54~55]	125**	Cancer
t_5	[54~55]	125**	Heart disease
t_6	[54~55]	125**	Tracheitis
t_7	[54~55]	125**	Cancer
t_8	[52~53]	125**	Tracheitis
t_9	[52~53]	125**	Tracheitis
t_{10}	[52~53]	125**	Tracheitis

Table 3 Result meeting 3-diversity

表 3 满足 3-多样模型的结果

	Age	ZipCode	Disease
t_1	[50~52]	125**	Heart disease
t_2	[50~52]	125**	Cancer
t_8	[50~52]	125**	Tracheitis
t_3	[51~55]	1253*	Heart disease
t_7	[51~55]	1253*	Cancer
t_9	[51~55]	1253*	Tracheitis
t_4	[53~54]	1255*	Cancer
t_5	[53~54]	1255*	Heart disease
t_6	[53~54]	1255*	Tracheitis
T_{10}	[53~54]	1255*	Tracheitis

实现匿名保护一般要对原始数据在准标识符上的属性值作数据概化,使得处理后的数据能够满足特定匿名模型的要求.因为 *k*-匿名模型存在隐私泄漏的风险,本文将主要研究如何通过数据概化来处理数据,使其满足 *l*-多样模型,同时尽可能地减少信息损失.我们下面将先给出本文采取的数据概化策略以及信息损失的量化定义,然后提出一种基于聚类的数据匿名方法.该方法不仅可以保证结果数据满足 *l*-多样模型的要求,而且它所带来的信息损失要远小于传统的满足 *l*-多样模型的数据匿名方法.

3 数据概化及信息损失

3.1 数据概化

数据概化的基本思想是用相同的抽象属性值来代替多个元组中原有的不同属性值.例如,表 1 中元组 t_4,t_5,t_6,t_{10}在属性 Age 上的值分别为 54,54,54,53,在表 3 中将其概化为一个相同的抽象区间值[53~54].通过数据概化来实现数据匿名的好处在于:虽然概化处理使得数据精确性有所降低,但在一定程度上可以保留数据原有的语义信息.

实现匿名保护往往需要经过多步数据概化,直到结果数据集满足特定匿名模型的要求.许多研究工作[3,7−9,13]采取一种基于概念层次结构的数据概化策略,它要求数据概化必须按照在属性值域上预先定义的概念层次(也称为 domain generalization hierarchy),自底向上逐步用抽象属性值代替原有属性值.这种数据概化策略限制了数据概化的选择余地,会产生不必要的信息损失.例如,针对表 1 的数据,图 1(a)和图 1(b)表示定义在属性 Age 上的概念层次,图 1(c)和图 1(d)表示定义在属性 ZipCode 上的概念层次.以属性 Age 为例,图 1(a)中 A_0 表示其原始值域,A_1 和 A_2 依次表示其更为抽象的概化值域.相应地,数据概化必须按照图 1(b)的方式自底向上进行.设两个元组的 Age 属性值分别为 51 和 52,为了使它们在概化后具有相同的属性值,必须将其均概化为[50~55].而如果不受概念层次的限制,可以将其概化为[51~52],从而保留更多的信息.

特别地,当采取基于概念层次结构的数据概化策略时,如果要求数据概化后同一属性的所有属性值具有相同的抽象程度(即处于同一概念层),则也称之为全值域概化(full-domain generalization)[9].全值域概化虽然实现

简单,但却会带来更多不必要的信息损失.例如,表 4 是对表 1 按照图 1 定义的概念层次作全值域概化的结果.虽然表 4 也满足 3-多样模型,但它使得所有元组只形成一个最大等价群.与表 3 相比,显然表 4 在 Age 和 ZipCode 属性上带来了更多的信息损失.

为了消除概念层次结构的限制,文献[12]提出将属性的值域划分为若干区域,然后将落在同一区域内的原有属性值用一个表示该区域的区间值来代替.这种数据概化策略要求一个属性的所有属性值之间存在全序关系,而实际中,有些属性的属性值之间并不存在这种关系.

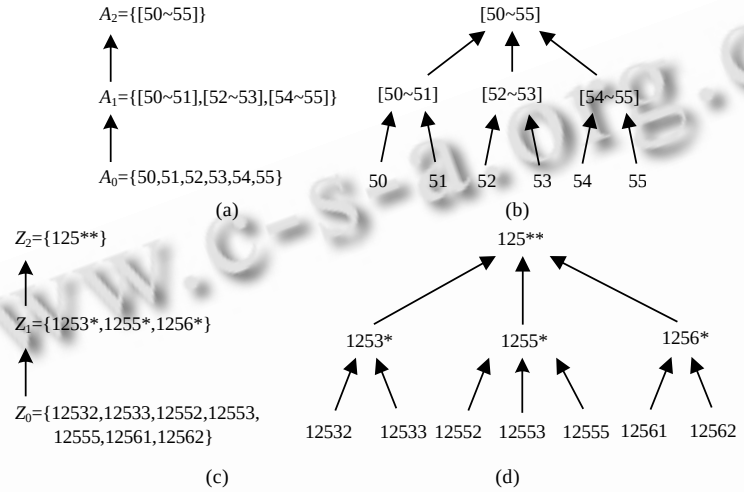


Fig.1 Domain generalization hierarchy

图 1 概念层次结构

Table 4 Result meeting 3-diversity (full-domain generalization)

表 4 满足 3-多样模型的结果(全值域概化)

	Age	ZipCode	Disease
t_1	[50~55]	125**	Heart disease
t_2	[50~55]	125**	Cancer
t_3	[50~55]	125**	Heart disease
t_4	[50~55]	125**	Cancer
t_5	[50~55]	125**	Heart disease
t_6	[50~55]	125**	Tracheitis
t_7	[50~55]	125**	Cancer
t_8	[50~55]	125**	Tracheitis
t_9	[50~55]	125**	Tracheitis
t_{10}	[50~55]	125**	Tracheitis

本文提出针对准标识符中不同类型的属性采取不同的数据概化策略.具体来说,我们将属性分为有序属性和无序属性两种类型.有序属性的属性值之间存在自然的顺序关系,例如年龄、身高、受教育程度高低等属性;无序属性的属性值之间不存在自然的顺序关系,如国籍、婚姻状况等.

对于有序属性,我们使用一个区间值$[a\sim b](a\le b)$来表示其属性值.特别地,对于原始属性值来说,有 $a=b$.设有 m 个元组 $t_i(i=1,2,\ldots,m)$,元组 t_i 在有序属性 $X\in QI$ 上的属性值为 $t_i[X]=[a_i\sim b_i]$.如果要求这 m 个元组概化后在 X 上具有相同属性值,则将 $t_i(i=1,2,\ldots,m)$在 X 上的取值都用新的区间值$[\mathrm{Min}(a_i)\sim\mathrm{Max}(b_i)]$来代替.例如,表 3 中将 t_4,t_5,t_6,t_{10} 在属性 Age 上的取值都用[53~54]来代替.

对于无序属性,我们使用一个集合值 y 来表示其属性值.特别地,对于原始属性值来说,y 为包含一个元素的集合.设有 m 个元组 $t_i(i=1,2,\ldots,m)$,元组 t_i 在无序属性 $Y\in QI$ 上的属性值为 $t_i[Y]=y_i$(y_i 为一个集合).如果要求这 m 个元组概化后在 Y 上具有相同属性值,则将 $t_i(i=1,2,\ldots,m)$在 Y 上的取值均用新的集合值 $\bigcup_{1\le i\le m}y_i$ 来代替.例如,设有 3 个元组在属性 Nationality 上的取值分别为{China},{India}和{China},可以用{China,India}来作为它们在

概化后的 Nationality 属性值.

显然,利用本节给出的针对有序属性和无序属性的概化策略,可以使多个元组经概化处理后在准标识符上具有相同的属性值,从而形成一个等价群.设有 m 个元组 $t_i(i=1,2,\ldots,m)$,采用上述概化策略处理后得到的等价群为 EQ,则 EQ 的代表元记为 $t_g=\varphi(t_1,t_2,\ldots,t_m)$,其中,对于有序属性 $X\in QI$,有 $t_g[X]=[\mathrm{Min}(a_i)\sim\mathrm{Max}(b_i)]$(设 $t_i[X]=[a_i\sim b_i]$,$i=1,2,\ldots,m$);对于无序属性 $Y\in QI$,有 $t_g[Y]=\bigcup_{1\le i\le m}y_i$ (设 $t_i[Y]=y_i,i=1,2,\ldots,m$);对于敏感属性 S,有 $t_g[S]=Null$.此外,从上述概化策略可以看出,有 $\varphi(t_1,t_2,\ldots,t_m)=\varphi(\ldots\varphi(\varphi(t_1,t_2),t_3)\ldots,t_m)$.

若无特殊说明,本文余下部分所提到的概化均采取本节所给出的针对有序属性或无序属性的概化策略.

3.2 信息损失

数据概化使得准标识符上属性值的精确性有所降低,会带来一定的信息损失.本文通过考察数据概化前后属性值不确定性程度的变化,来量化地定义信息损失的大小.具体来说,对于有序属性和无序属性,数据概化带来的信息损失分别定义如下:

定义 6(概化有序属性的信息损失). 设元组 t 在有序属性 X 上的值为 $x=[a\sim b]$,概化后的相应属性值为 $x^*=[a^*\sim b^*]$,则对于元组 t 来说,其在有序属性 X 上的信息损失为

$$L(x,x^*)=\begin{cases}(b^*-a^*+1)/(b-a+1), & x\neq x^*\\ 0, & x=x^*\end{cases}.$$

定义 7(概化无序属性的信息损失). 设元组 t 在无序属性 Y 上的值为集合 y,概化后的相应属性值为集合 y^*,则对于元组 t 来说,其在无序属性 Y 上的信息损失为

$$L(y,y^*)=\begin{cases}|y^*|/|y|, & y\neq y^*\\ 0, & y=y^*\end{cases},$$

其中,$|y|$和$|y^*|$分别表示概化前后属性值集合 y 和 y^*中元素的个数.

在此基础上,我们可以进一步定义数据概化在特定元组上带来的信息损失,以及在整个数据集上带来的信息损失.

定义 8(概化元组的信息损失). 设准标识符为 QI,如果元组 t 经概化处理后为 t^*,则将 t 概化为 t^*所产生的信息损失为

$$L(t,t^*)=\sum_{i=1}^{u}L(t[X_i],t^*[X_i])+\sum_{j=1}^{v}L(t[Y_j],t^*[Y_j]),$$

其中,$X_i(i=1,2,\ldots,u)$,$Y_i(j=1,2,\ldots,v)$分别为 QI 中的有序属性和无序属性,$t[X_i]$,$t[Y_j]$和 $t^*[X_i]$,$t^*[Y_j]$分别为 t 和 t^*在属性 X_i 和 Y_j 上的取值.

考虑到准标识符中不同属性在数据共享中的重要性可能有所不同,在计算信息损失时可以为不同属性定义不同的权重.权重值较大的属性具有较高的重要性,在数据概化时要尽可能地保持该属性值的精确性.

定义 9(概化元组的加权信息损失). 设准标识符为 QI,如果元组 t 经概化处理后为 t^*,则将 t 概化为 t^*所产生的信息损失为

$$L(t,t^*)=\sum_{i=1}^{u}p_i\times L(t[X_i],t^*[X_i])+\sum_{j=1}^{v}q_j\times L(t[Y_j],t^*[Y_j]),$$

其中,$X_i(i=1,2,\ldots,u)$,$Y_i(j=1,2,\ldots,v)$分别为 QI 中的有序属性和无序属性,$t[X_i]$,$t[Y_j]$和 $t^*[X_i]$,$t^*[Y_j]$分别为 t 和 t^*在属性 X_i 和 Y_j 上的取值,$p_i(p_i\ge 0)$和 $q_j(q_j\ge 0)$分别为属性 X_i 和 Y_j 的权重.

定义 10(概化数据集的信息损失). 设原始数据集为 D,概化后的结果数据集为 D^*,则将 D 概化为 D^*所产生的信息损失为

$$L(D,D^*)=\sum_{i=1}^{n}L(t_i,t_i^*),$$

其中,$t_i(i=1,2,\ldots,n)$为 D 中元组,t_i^* 为 t_i 在 D^*中的对应概化元组.

定义 11(完全概化数据集). 设原始数据集为 D,准标识符为 QI.如果将 D 在 QI 上概化为 D_c,使得 D_c 中所有元组在 QI 上取值都相同,则称 D_c 为 D 的完全概化数据集.

注意到,给定原始数据集 D 和准标识符 QI,当采取本文第 3.1 节给出的数据概化策略时,相应的完全概化数据集具有唯一性.

定义 12(概化数据集的相对信息损失). 设 D 为原始数据集,QI 为准标识符,D^*为 D 在 QI 上的一个概化数据集,D_c 为 D 在 QI 上的完全概化数据集,则将 D 概化为 D^*的相对信息损失为

$$RL(D,D^*)=\frac{L(D,D^*)}{L(D,D_c)}\times 100\%.$$

由于当给定原始数据集 D 和准标识符 QI 时,完全概化数据集 D_c 具有唯一性,所以在实际算法中,我们只需考虑 $L(D,D^*)$.而 $RL(D,D^*)$主要用于在第 5 节的实验结果中以直观形式反映结果数据集的相对信息损失程度.

4 基于聚类的数据匿名方法

本节将给出一种数据匿名方法,其基于聚类的思想来寻找合适的概化方案,可以满足 l-多样模型的匿名保护要求,并降低数据概化带来的信息损失.该方法的基本思想是:首先对原始数据集中的元组作聚类,使得每个类内的元组在准标识符上具有尽可能相似的属性值,同时在敏感属性上至少有 l 个不同的值;然后将同一个类中的所有元组作概化处理,使其在准标识符上具有相同的概化属性值,形成一个等价群.这样,就得到满足 l-多样模型的结果数据集.

数据匿名问题可以看作是带特定约束的聚类问题,即必须满足匿名模型的约束要求.但其与一般聚类问题有明显区别,使得传统聚类方法并不适合直接用于解决数据匿名问题.一般聚类问题要求在聚类结果中,类内对象尽可能地相似,而类间对象尽可能地互不相似[18].而数据匿名问题则强调首先要满足匿名模型的要求,然后,在此前提下考虑使得聚类结果尽可能地有助于降低后续概化处理带来的信息损失.

具体到本文所考虑的 l-多样模型来说,首先是要求在聚类结果中,每个类内的元组在敏感属性上至少需要有 l 个不同的值;其次才要求同一个类内的元组在准标识符上具有相似的属性值,以便降低后续概化处理带来的信息损失.下面结合 l-多样模型给出一个实例,说明传统聚类方法不适合直接用于解决数据匿名问题.

设表 5(a)是一个病人诊断记录的数据集,其中,Disease 为敏感属性,属性组(Age,ZipCode)为准标识符.要求对其在准标识符上的属性值作概化处理,以满足 2-多样模型(即 l=2).如果使用传统的聚类方法,对元组在准标识符上的属性值作聚类,根据类内对象尽可能相似、而类间对象尽可能互不相似的原则[18],表 5(a)中的元组可被聚为两个类:$\{t_1,t_4\}$和$\{t_2,t_3\}$.此时,虽然每个类内的元组无需概化处理就已经形成一个等价群,但是这样的结果并不能满足 2-多样模型(见表 5(b)).而实际能够满足 2-多样模型的概化数据集见表 5(c).

Table 5 Difference between data anonymization and general clustering

表 5 数据匿名问题与一般聚类问题的不同

(a)

	Age	ZipCode	Disease
t_1	51	12320	Heart disease
t_2	56	12320	Cancer
t_3	56	12320	Cancer
t_4	51	12320	Heart disease

(b)

	Age	ZipCode	Disease
t_1	51	12320	Heart disease
t_4	51	12320	Heart disease
t_2	56	12320	Cancer
t_3	56	12320	Cancer

(c)

	Age	ZipCode	Disease
t_1	[51~56]	12320	Heart disease
t_2	[51~56]	12320	Cancer
t_3	[51~56]	12320	Cancer
t_4	[51~56]	12320	Heart disease

本节余下部分将首先给出相关的距离定义,然后在传统的层次聚类基础上加以改造,给出可以满足 l-多样模型约束要求的数据匿名具体算法实现,并对算法的正确性和复杂性加以分析.

4.1 距离定义

考虑到满足匿名保护需求的概化数据集可以有多个,它们对应的信息损失可能各不相同.本文考虑从信息损失的角度出发,来定义数据对象之间的距离.

定义 13(元组间距离). 设有元组 t_1 和 t_2,如果 $t^*=\varphi(t_1,t_2)$为将 t_1 和 t_2 概化后所形成等价群的代表元,则 t_1 与 t_2

之间的距离为

$$DS(t_1,t_2)=L(t_1,t^*)+L(t_2,t^*),$$

其中,$L(t_1,t^*)$和 $L(t_2,t^*)$分别为将 t_1 和 t_2 概化成 t^*所导致的信息损失(见定义 8 和定义 9).

定义 14(类的代表元组). 设有类 G.将 G 中所有元组作概化处理,使之形成等价群 EQ.如果 t_g 为 EQ 的代表元,则也称 t_g 为类 G 的代表元组.

定义 15(元组到类的距离). 设 t_g 为类 G 的代表元组,如果元组 $t\notin G$,则 t 到类 G 的距离为

$$DS(t,G)=DS(t,t^*)+|G|\times DS(t_g,t^*),$$

其中,$|G|$为类 G 中包含元组的数目,$t^*=\varphi(t,t_g)$为将 t 和 t_g 概化后所形成等价群的代表元.

定义 16(类间距离). 设有两个类 G 和 H,其代表元组分别为 t_g 和 t_h,则类 G 与 H 之间的距离为

$$DS(G,H)=|G|\times DS(t_g,t^*)+|H|\times DS(t_h,t^*),$$

其中,$|G|$和$|H|$分别为类 G 和 H 中包含元组的数目,$t^*=\varphi(t_g,t_h)$为将 t_g 和 t_h 概化后所形成等价群的代表元.

根据上述定义,在计算距离时,只需先使用第 3.1 节给出的针对有序属性和无序属性的概化策略临时作试探性概化处理,得到概化后所形成等价群的代表元 t^*,然后就可以计算出相应的距离,而不需要对所有元组都作真实的概化处理.以计算两个类 G 和 H 间的距离为例(设 G 和 H 的代表元组分别为 t_g 和 t_h),只需先计算出 $t^*=\varphi(t_g,t_h)$,即对 G 和 H 的代表元组临时作试探性概化处理,然后可依据定义 16 计算出 G 和 H 间的距离.

4.2 算 法

本节给出实现数据匿名的具体算法,记为 L-clustering.其对传统的凝聚层次聚类(agglomerative hierarchical clustering)加以改造,在保证最后的结果数据集可以满足 l-多样模型的前提下来寻找合适的概化方案,以减少实现数据匿名时的信息损失.具体的算法细节如下所示:

算法. L-clustering.

输入:原始数据集 D,准标识符 QI,敏感属性 S,l-多样模型参数 $l(l>1)$;

输出:结果数据集 D^*.

过程:

1. $Q=\varnothing$;
2. 如果 D 中不同敏感属性值的个数小于 l,返回.
3. 当 D 中不同敏感属性值个数不小于 l 时,循环执行:
   a. 随机地从 D 中选取元组 t,$D=D-\{t\}$;
   b. 生成类 $G=\{t\}$;$t_g[QI]=t[QI]$,$t_g[S]=Null$; /*t_g 为类 G 的代表元组*/
   c. 当 G 中元组个数少于 l 时,循环执行:
      (1) $t'=\arg\min_{(t_i\in D)\wedge(\forall t_j\in G:t_i[S]\neq t_j[S])}(DS(t_i,G))$;
      (2) $G'=\arg\min_{G_k\in Q}(DS(G_k,G))$;
      (3) 若 $DS(t',G)\leq DS(G',G)$,则 $D=D-\{t'\},G=G\cup\{t'\},t_g=\varphi(t_g,t')$;
      否则,$Q=Q-\{G'\},G=G\cup G',t_g=\varphi(t_g,t_{g'})$; /*$t_{g'}$ 为类 G'的代表元组*/
   d. $Q=Q\cup\{G\}$;
4. 当 D 中仍有剩余元组时,循环执行:
   a. 随机地从 D 中选取元组 t',$D=D-\{t'\}$;
   b. $G=\arg\min_{G_k\in Q}(DS(t',G_k))$;
   c. $G=G\cup\{t'\},t_g=\varphi(t_g,t')$; /*$t_g$ 为类 G 的代表元组*/
5. 依次处理 Q 中的每个类,将类中的每个元组在准标识符上的属性值用该元组所在类的代表元组中的相应值代替,得到结果数据集 D^*.

算法 L-clustering 首先判断输入数据集 D 中不同敏感属性值的个数是否大于等于 l(步骤 2).如果这一条件

不能满足,则肯定不能产生一个符合 l-多样模型的结果数据集,算法返回.

当 D 中不同敏感属性值的个数不小于 l 时,算法重复执行步骤 3,每次得到一个类 G,使得 G 中至少有 l 个元组具有互不相同的敏感属性值.具体方法是,先随机地从 D 中选取元组 t 来初始化类 G,然后在步骤 3.c 采取贪婪法,每次从 D 中选取距 G 最近的元组 t',并要求 t' 与 G 中已有元组的敏感属性值均不相同.同时,从已生成类的集合 Q 中选取距 G 最近的类 G'(注意到 Q 中的任意一个类都已经至少有 l 个元组具有互不相同的敏感属性值).如果 $DS(t',G)\le DS(G',G)$,则将 t' 从 D 中删除并加入到 G 中;否则从 Q 中删除 G',并将 G' 合并到 G 中.这一过程重复执行,直到 G 中有不少于 l 个元组为止.然后,将 G 加入到 Q 中.

在步骤 3 结束后,如果 D 中仍有剩余元组,算法在步骤 4 将剩余的元组逐一加入到距其最近的已有类中.最后,算法在步骤 5 将每个元组在准标识符上的属性值用其所在类的代表元组在准标识符上的属性值来代替,得到满足 l-多样模型的数据集 D^*.

算法在步骤 3.c 和步骤 4.b 中,根据第 4.1 节给出的定义来计算元组与类以及类与类之间的距离(即算法中对 DS 的计算).为了方便距离计算,在算法执行过程中为每个类 G 保存其代表元组 t_g.初始时(步骤 3.b),每个类中只包含一个元组 t,此时,其代表元组为 $t_g=(t[QI],Null)$.在算法的执行过程中,对类 G 的代表元组作增量式的更新计算:当有新的元组 t' 加入到类 G 中,相应地更新 G 的代表元组为 $t_g=\varphi(t_g,t')$ (步骤 3.c.(3)和步骤 4.c);当有另外一个类 G' 需要合并入类 G 时,相应地更新 G 的代表元组为 $t_g=\varphi(t_g,t_{g'})$ (步骤 3.c.(3)),其中,$t_{g'}$ 为类 G' 的代表元组.根据第 3.1 节给出的针对有序属性和无序属性的概化策略,有 $\varphi(t_1,t_2,...,t_m)=\varphi(\ldots\varphi(\varphi(t_1,t_2),t_3)...,t_m)$,故可以保证上述对类的代表元组作增量式更新计算的正确性.

4.3 算法的正确性和复杂性分析

4.3.1 正确性分析

不难看出,如果 D 中不同敏感属性值的个数不小于 l,则 L-clustering 输出的结果数据集可以保证满足 l-多样模型的要求.实际上,当步骤 3 完成时,算法已经得到类的集合 Q,使得 Q 中的每个类至少有 l 个元组,它们的敏感属性值各不相同.而随后的步骤 4 继续保持了这一结果,直至 D 中余下的每个元组都被归入一个类中.最后,步骤 5 实际上将每个类概化成为一个等价群,从而得到满足 l-多样模型的输出数据集 D^*.

4.3.2 复杂性分析

设原始数据集 D 中的元组数目为 $|D|=n$,准标识符维数为 $|QI|=d$,算法在步骤 3 完成后得到 $|Q|=m$ 个类.不难看出,有 $1\le m\le n/l (l>1)$.

算法在步骤 2 检查 D 中不同敏感属性值的个数.这一步只需扫描 D 一次,执行时间为 $O(n)$.

算法在步骤 3 中,每得到一个新类 G,至多 $l-1$ 遍扫描 D 和 Q,并计算在准标识符 QI 上的相应距离.因为在算法执行中有 $|D|+|Q|\le n$,所以每次生成一个类的时间不超过 $O(dln)$.共生成 m 个类,因此,步骤 3 的执行时间为 $O(dlmn)$.

因为步骤 3 结束时得到 m 个类,每个类中至少有 l 个元组,故至多还剩余 $n-ml$ 个元组.因此,算法步骤 4 的执行次数至多为 $n-ml$.每次循环要扫描 Q 一遍,并计算在准标识符 QI 上的相应距离,执行时间为 $O(dm)$.所以,步骤 4 的执行时间为 $O(dm(n-ml))$.

算法在步骤 5 生成结果数据集时,需扫描所有元组一次,同时替换元组在准标识符上的属性值,故执行时间为 $O(dn)$.

因此,L-clustering 总体执行时间为 $O(n)+O(dlmn)+O(dm(n-ml))+O(dn)=O(dlmn)$.因为 $lm<n$,所以在最坏情况下,L-clustering 的时间复杂度为 $O(dn^2)$.

5 实验结果

在这一节中,我们通过实验分析 L-clustering 的性能,并将其与文献[3]提出的实现 l-多样模型的数据匿名方法(为了方便起见,将其标记为 L-incognito)进行比较.L-incognito 递增地检查准标识符的子属性集.对于每个子

属性集,自底向上采取全值域概化策略,直到概化结果在当前子属性集上满足 *l*-多样模型.然后,从满足条件的低维子属性集上的概化值出发,组合得到高维子属性集上的候选者,并再次自底向上作全值域概化.重复这一过程,直至概化结果在整个准标识符上满足 *l*-多样模型.

实验所使用的数据集为 UCI 机器学习数据库中的 Adult 数据集.该数据集包括部分美国人口普查数据,在数据匿名保护研究中被广泛使用,已成为该领域事实上的标准测试数据集.与文献[3]一样,实验时包含缺失值的数据记录被删除,经处理后的数据集共有 45 222 个元组;同时只保留 9 个属性:Age,Gender,Race,Marital Status,Education,Native Country,Work Class,Salary Class,Occupation.其中,Age 属性为有序属性,其他属性均为无序属性.实验中,分别取前 d(d=2,…,8)个属性形成准标识符.在计算信息损失时,准标识符中各属性的权重均设为 1.同时,将属性 Occupation 作为敏感属性,实验数据集中不同 Occupation 属性值的个数为 14.实验结果图中所用标记的含义见表 6.实验的硬件环境为 Intel Pentium IV 2.8GHz CPU,512MB RAM,操作系统为 Microsoft Windows XP.在实验中,*L*-clustering 和 *L*-incognito 均使用 C++实现.

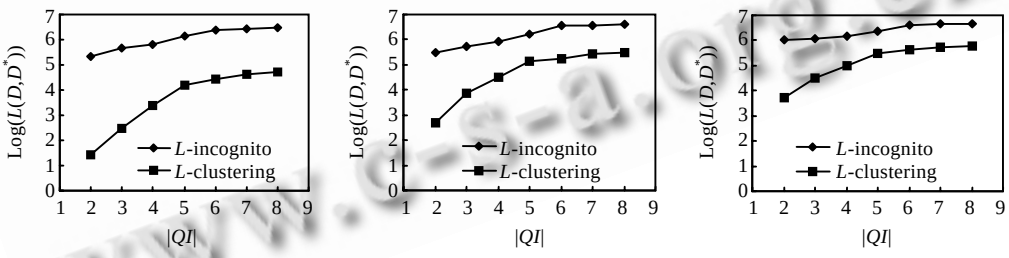
Table 6 List of symbols in experimental results

表 6 实验结果中所用符号的含义

Symbol	Meanings in experimental results
D	Original dataset
D^*	Resultant dataset
QI	Quasi-Identifier
l	Parameter for l-diversity model
$\|QI\|$	Number of attributes in quasi-identifier QI
$\|D\|$	Number of tuples in dataset D

5.1 信息损失分析

图 2(a)~图 2(c)分别给出了当 l=2,l=7 和 l=12 时,准标识符维数$|QI|$变化对 *L*-clustering 和 *L*-incognito 的信息损失大小的影响.为了便于表示,图中信息损失用 $L(D,D^*)$的对数形式来表示,其中,$L(D,D^*)$的含义见定义 10.可以看到,当准标识符维数$|QI|$增加时,*L*-incognito 和 *L*-clustering 的信息损失均随之而增加.这是由于当 l 值相同时,随着$|QI|$的增加,每个元组需在更多属性上作概化.因此,信息损失也相应增大.



(a) l=2, $|D|$=45222 (b) l=7, $|D|$=45222 (c) l=12, $|D|$=45222

Fig.2 Information loss when varying $|QI|$

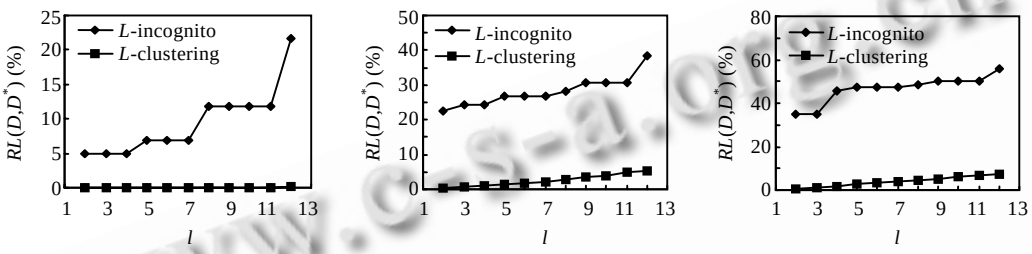
图 2 准标识符维数$|QI|$变化下的信息损失

但在同等条件下(即当 l 和$|QI|$均相同时),*L*-incognito 产生的信息损失要远大于 *L*-clustering.当 l 和$|QI|$均较小时,这一现象尤为明显.例如,当 l=2,$|QI|$=2 时(图 2(a)),*L*-incognito 的信息损失约为 *L*-clustering 的 8 000 倍.这是因为 *L*-incognito 使用全值域概化策略,全值域概化策略通常容易产生不必要的过度概化,因而带来的信息损失较大;而 *L*-clustering 采取更为灵活的数据概化策略,不受概念层次结构限制,能够找到具有较小信息损失的概化方案来满足 *l*-多样模型的匿名保护要求.

图 3(a)~图 3(c)分别给出了当$|QI|$=2,$|QI|$=5 和$|QI|$=8 时,*l*-多样模型的参数 l 值变化对 *L*-clustering 和 *L*-incognito 的信息损失大小的影响.考虑到当原始数据集和准标识符确定时,完全概化数据集具有唯一性.图中纵坐标用更为直观的相对信息损失 $RL(D,D^*)$来表示($RL(D,D^*)$的含义见定义 12).随着 l 值的增大,*L*-clustering 和

L-incognito 的信息损失均有所增加.这是因为随着 *l* 值的增加,*l*-多样模型要求结果数据集中每个等价群包含更多的元组(至少 *l* 个元组).因此,信息损失也相应增大.

但从图 3 中可以看到,*L*-incognito 的信息损失在同等条件下要大于 *L*-clustering.此外,随着 *l* 值的增加,*L*-clustering 的信息损失只是缓慢增长,而 *L*-incognito 的信息损失则呈跳跃式分阶段增长.这一情况在|*QI*|较小时尤为明显.例如图 3(a)显示,当 *l* 从 8 增加到 11 时,*L*-incognito 产生同样的信息损失;而当 *l*=12 时,其信息损失陡然增加.产生这种现象是由于 *L*-incognito 采取全值域概化策略,经常会导致过度概化,带来不必要的信息损失.实际上,在图 3(a)中,当 *l*=8 时,*L*-incognito 已将原始数据集过度概化,使得处理结果可以满足 11-多样模型了.

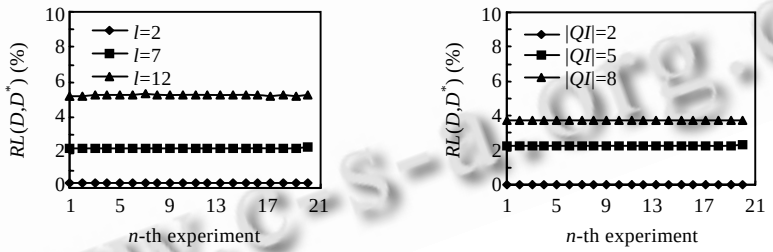


(a) |*QI*|=2, |*D*|=45222 (b) |*QI*|=5, |*D*|=45222 (c) |*QI*|=8, |*D*|=45222

Fig.3 Information loss when varying the value of *l*

图 3 *l* 值变化下的信息损失

图 4 考察 *L*-clustering 中随机选择元组(见算法 *L*-clustering 中的步骤 3.a 和步骤 4.a)而引入的随机性对信息损失的影响.图中给出了对于多组给定的准标识符维数|*QI*|和 *l*-多样模型的参数 *l* 值,重复 20 次执行 *L*-clustering 的实验结果.图中横坐标表示是第 *n*(*n*=1,2,…,20)次实验,纵坐标为相对信息损失 $RL(D,D^*)$ ($RL(D,D^*)$的含义见定义 12).从实验结果中可以看出,|*QI*|和 *l* 是影响信息损失变化的主要因素;而 *L*-clustering 中随机选择元组的方式对信息损失的稳定性影响相对较小.



(a) |*QI*|=5, |*D*|=45222 (b) *l*=7, |*D*|=45222

Fig.4 Effect of *L*-clustering's randomization on information loss

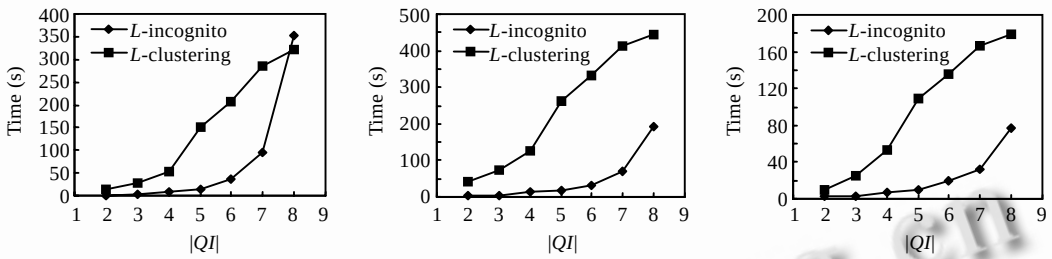
图 4 *L*-clusteing 中的随机性对信息损失的影响

5.2 执行时间分析

图 5 给出了当 *l* 值固定时,准标识符维数|*QI*|变化对 *L*-incognito 和 *L*-clustering 执行时间的影响.随着|*QI*|的增加,两者的执行时间都有所增加.但是,*L*-incognito 的执行时间增长呈明显加速趋势.由于 *L*-incognito 通过递增地考察准标识符子属性集上的概化属性值组合来寻找可实现匿名保护的全值域概化方案,在最坏情况下,*L*-incognito 的执行时间随着准标识符维数增加将呈指数式增长.尤其是当 *l* 取值较小时(如图 5(a)所示),在准标识符子属性集上满足 *l*-多样模型的概化属性值组合数目较多,所以 *L*-incognito 的执行时间随着|*QI*|的增加而显著地增加;而 *L*-clustering 通过考察元组与类之间以及类与类之间的距离寻找合适的概化方案,以较小的信息损失来满足匿名保护的需求,其执行时间随着准标识符维数的增加而呈线性增长趋势.

图 6 给出了当准标识符维数|*QI*|固定时,*l* 值变化对 *L*-clustering 和 *L*-incognito 执行时间的影响.对于 *L*-incognito 来说,当|*QI*|较小时(如图 6(a)所示),其执行时间随 *l* 值的增加而呈增加趋势;当|*QI*|较大时,其执行时间随 *l* 值的增加而呈减少趋势(如图 6(c)所示);而当|*QI*|介于两者之间时,其执行时间呈波动趋势(如图 6(b)所示).这种现象是由两方面因素所造成的:一方面,*L*-incognito 在准标识符 *QI* 的每个子属性集上采取自底向上的概化策略.随着 *l* 值的增加,它需要作更多次概化尝试,直到概化处理结果满足 *l*-多样模型的需求.所以,*l* 值增加使其执行时间有增加的趋势.另一方面,增加 *l* 值,使得低维子属性集上满足 *l*-多样模型的概化属性值数目减少,组合后得到高维子属性集上的候选者数目也随之减少.因此,随着 *l* 值的增加,*L*-incognito 的执行时间又有减少趋势.尤其是当|*QI*|较大时,原来的许多候选概化属性值组合随着 *l* 值的增加将不再满足 *l*-多样模型.此时,这种执行时间减少的趋势在多数情况下将占主导地位.

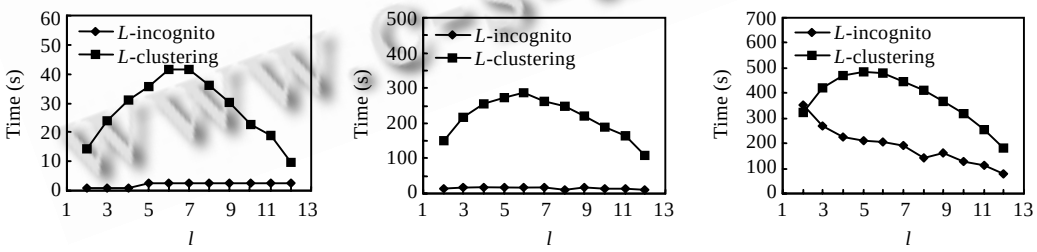
对于 *L*-clustering 来说,其执行时间随着 *l* 值的增加表现为先增加而后减少的状况.因为在 *L*-clustering 得到的每个类中,第 1 个元组是随机选取的,所花费的时间较短.而随后加入元组到类中,则需要多次进行距离计算来找到距离最小的元组或类,因此所花费的时间较长.而当 *l* 增加时,*L*-clustering 产生的类的数目将减少;特别是当 *l* 较小时(例如当 *l* 在 2~5 之间),类的数目将随 *l* 的增加而显著减少.这意味着在聚类时,更多的元组需要进行多次距离计算.因此,当 *l* 较小时,*L*-clustering 的总体执行时间随着 *l* 值的增加而增加.与此同时,因为 *L*-clustering 要求类中元组在敏感属性上至少有 *l* 个不同值.随着 *l* 的增加,可满足 *l*-多样模型的元组组合数目减少.特别地,当 *l* 较大时,*L*-clustering 通过敏感属性值是否相同就可以排除大多数元组,将会减少距离计算的时间.因此,当 *l* 较大时(例如当 *l* 在 7~12 之间),随着 *l* 值的进一步增加,*L*-clustering 的执行时间反而减少.



(a) *l*=2, |*D*|=45222 (b) *l*=7, |*D*|=45222 (c) *l*=12, |*D*|=45222

Fig.5 Running time when varying |*QI*|

图 5 准标识符维数|*QI*|变化下的执行时间



(a) |*QI*|=2, |*D*|=45222 (b) |*QI*|=5, |*D*|=45222 (c) |*QI*|=8, |*D*|=45222

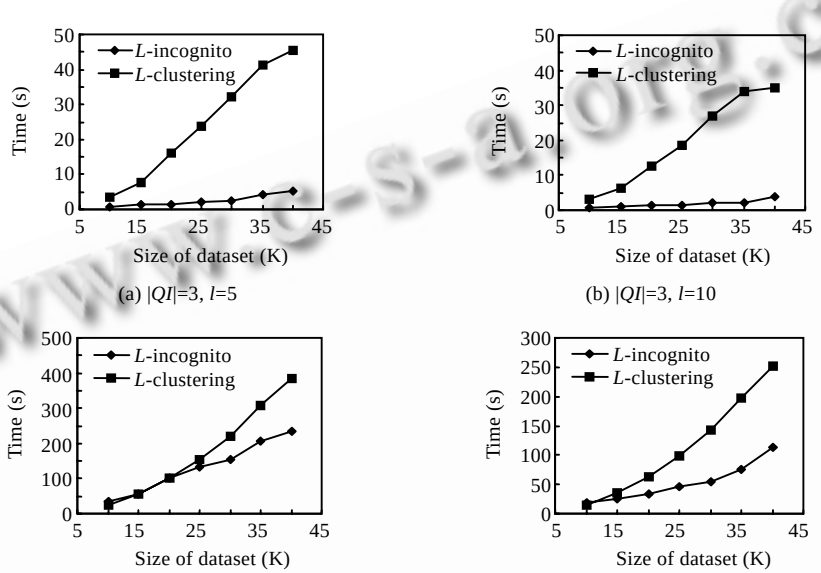
Fig.6 Running time when varying the value of *l*

图 6 *l* 值变化下的执行时间

图 7 给出了当|*QI*|和 *l* 值固定时,数据集大小|*D*|对两种算法执行时间的影响.图中数据集大小单位为 K,即 1 000 个元组.实验时,从原始数据集中随机抽取 10K,15K,…,40K 个元组,形成不同大小的数据集.从图中可以看到,由于 *L*-clustering 不受概念层次结构限制、采取更为灵活的数据概化策略,使得可供选择的概化方案数目大为增加.它需要通过计算元组与类之间以及类与类之间的距离来寻找合适的概化方案,以减少实现匿名保护时

的信息损失,因此花费时间较长.其执行时间在最坏情况下与数据集中元组数目成平方关系.但是,考虑到匿名保护通常是在数据共享前的一个离线处理过程,并不需要实时的处理能力.因此,*L*-clustering 的执行时间还是可以接受的.

L-incognito 采取全值域概化策略,其执行时间与数据集大小之间呈线性增长关系.但是,它需要递增地在准标识符的子属性集上作数据概化.因此,在最坏情况下,该线性增长关系的系数为 $2^{|QI|}$.这使得当|*QI*|较大时,*L*-incognito 的执行时间有时甚至长于 *L*-clustering.例如,当|*QI*|=8,*l*=5,|*D*|=10000 时(如图 7(c)所示),可以看到,*L*-incognito 的执行时间比 *L*-clustering 要长.



(a) |*QI*|=3, *l*=5 (b) |*QI*|=3, *l*=10

(c) |*QI*|=8, *l*=5 (d) |*QI*|=8, *l*=10

Fig.7 Running time when varying the size of dataset

图 7 数据集大小变化下的执行时间

6 结束语

针对数据共享中敏感属性的匿名保护,本文提出了一种基于聚类的数据匿名方法 *L*-Clustering.该方法可以保证结果数据集满足 *l*-多样模型,能够很好地实现数据共享中的匿名保护需求,防止与个体相关的敏感属性值的泄漏.同时,该方法消除了传统数据概化处理时的概念层次结构限制,采取更为灵活的数据概化策略;并基于聚类的思想来寻找合适的概化方案,在实现匿名保护时,可以有效地减少概化处理所带来的信息损失.今后的工作将考虑对寻找合适概化方案过程中的距离计算进行优化,以进一步提高算法的执行效率.

References:

[1] Samarati P. Protecting respondents' identities in microdata release. IEEE Trans. on Knowledge and Data Engineering, 2001,13(6): 1010−1027. [doi: 10.1109/69.971193]

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

[3] Machanavajjhala A, Gehrke J, Kifer D. *l*-Diversity: Privacy beyond *k*-anonymity. In: Liu L, Reuter A, Whang KY, Zhang JJ, eds. Proc. of the 22nd Int'l Conf. on Data Engineering. Los Alamitos: IEEE Computer Society, 2006. 24.

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

[5] Aggarwal G, Feder T, Kenthapadi K, Khuller S, Panigrahy R, Thomas D, Zhu A. Achieving anonymity via clustering. In: Vansummeren S, ed. Proc. of the 25th ACM SIGMOD-SIGACT-SIGART Symp. on Principles of Database Systems. New York: ACM, 2006. 153−162.

[6] Meyerson A, Williams R. On the complexity of optimal *k*-anonymity. In: Deutsch A, ed. Proc. of the 23rd ACM SIGACT-SIGMOD-SIGART Symp. on Principles of Database Systems. New York: ACM, 2004. 223−228.

[7] Iyengar V. Transforming data to satisfy privacy constraints. In: Zaïane O, ed. Proc. of the 8th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. New York: ACM, 2002. 279−288.

[8] Wang K, Yu P, Chakraborty S. Bottom-Up generalization: A data mining solution to privacy protection. In: Rastogi R, Morik K, Bramer M, Wu XD, eds. Proc. of the 4th IEEE Int'l Conf. on Data Mining. Los Alamitos: IEEE Computer Society, 2004. 249−256.

[9] LeFevre K, DeWitt D, Ramakrishnan R. Incognito: Efficient full-domain *k*-anonymity. In: Ozcan F, ed. Proc. of the 24th ACM SIGMOD Int'l Conf. on Management of Data. New York: ACM, 2005. 49−60.

[10] Fung B, Wang K, Yu P. Top-Down specialization for information and privacy preservation. In: Toyama M, Sasaki S, eds. Proc. of the 21st Int'l Conf. on Data Engineering. Los Alamitos: IEEE Computer Society, 2005. 205−216.

[11] Bayardo R, Agrawal R. Data privacy through optimal *k*-anonymization. In: Toyama M, Sasaki S, eds. Proc. of the 21st Int'l Conf. on Data Engineering. Los Alamitos: IEEE Computer Society, 2005. 217−228.

[12] LeFevre K, DeWitt D, Ramakrishnan R. Mondrian multidimensional *k*-anonymity. In: Liu L, Reuter A, Whang KY, Zhang JJ, eds. Proc. of the 22nd Int'l Conf. on Data Engineering. Los Alamitos: IEEE Computer Society, 2006. 25.

[13] Li JY, Wong R, Fu A, Pei J. Achieving *k*-anonymity by clustering in attribute hierarchical structures. In: Tjoa AM, Trujillo J, eds. Proc. of the 8th Int'l Conf. on Data Warehousing and Knowledge Discovery. Heidelberg: Springer-Verlag, 2006. 405−416.

[14] Byun JW, Kamra A, Bertino E, Li NH. Efficient *k*-anonymization using clustering techniques. In: Kotagiri R, Krishna PR, Mohania M, Nantajeewarawat E, eds. Proc. of the 12th Int'l Conf. on Database Systems for Advanced Applications. Heidelberg: Springer-Verlag, 2007. 188−200.

[15] Yang XC, Liu XY, Wang B, Ge Y. *K*-Anonymization approaches for supporting multiple constraints. 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]

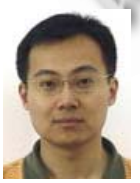
[16] Xu J, Wang W, Pei J, Wang XY, Shi BL, Fu A. Utility-Based anonymization using local recoding. In: Eliassi-Rad T, Ungar L, Craven M, Gunopulos D, eds. Proc. of the 12th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. New York: ACM, 2006. 785−790.

[17] Xiao XK, Tao YF. Anatomy: Simple and effective privacy preservation. In: Dayal U, Whang KY, Lomet D, Alonso G, Lohman G, Kersten M, Cha SK, Kim YK, eds. Proc. of the 32nd Int'l Conf. on Very Large Data Bases. New York: ACM, 2006. 139−150.

[18] Han J, Kamber M. Data Mining: Concepts and Techniques. 2nd ed., San Francisco: Morgan Kaufmann Publishers, 2006. 383−386.

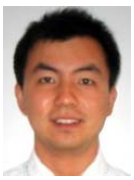
附中文参考文献:

[15] 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]



王智慧(1975－),男,安徽滁州人,博士,讲师,CCF 学生会员,主要研究领域为数据安全,数据挖掘,数据库.

汪卫(1970－),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为数据库,数据挖掘.



许俭(1983－),男,硕士,主要研究领域为数据安全,数据挖掘.

施伯乐(1935－),男,教授,博士生导师,CCF 高级会员,主要研究领域为数据库,知识库.