**数据发布中多敏感属性数据隐私保护方法研究**

**摘要**

**英文摘要**

**目录**

1. **绪论**

1.1数据发布研究背景及意义

随着互联网的高速发展，社会信息化和网络化的发展导致数据爆炸式增长，同时大量个人数据能够用数据计算的方法进行收集和分。大量数据信息存在蕴含了不可估量的信息价值，因此也导致了数据挖掘工具的广泛使用，各种公开未公开的用数据被各种分析挖掘其中的信息，在这些信息创造价值的同时，也使得人们对保护用户个人的隐私信息不被泄露，不被恶意使用泄露敏感信息等问题有

了极大的关注【1】。然而数据发布是在当前数据管理、数据挖掘与信息共享应用中等环节中很重要的一个部分，是指从大量的数据中通过算法搜索隐藏于其中信息，从大批数据中提取出潜在的、有价值的信息的过程，该过程效果是否能取得好的效果取决于是否有可用的高质量数据。信息共享是指按照一定的规则或者协议在多个数据持有者之间进行交换数据信息。除此之外，在实际生活场景

中，由于公共科学或公共服务等社会化需求，有很多机构需要定期对外发布数据。例如，医院定期发布的患者医疗统计数据，上市公司需要定期公布其公司的财务报表等。在当前网络信息技术如此发达的情况下，数据存储技术和个人或商用计算机都具备了高性能的处理能力后，海量数据的收集、公布和各种分析变得越来越简单易行。【2】与此同时，同样给数据的隐私保护问题带来了巨大的挑

战。比如，通过对医院发布的病患的医疗数据进行分析，可以发现各种疾病之间的关联性或分布区域性等特点，具有巨大的科学研究价值。但同时在分析的过程中，必然会涉及到患者的一些个人数据，从而可能造成病人包括疾病等敏感信息的泄露。因此，如何有效地解决数据发布过程中可能存在的隐私泄露问题，已经成为目前数据管理、数据挖掘和信息共享领域的一个研究热点，形成一个新的

研究领域——数据发布中的隐私保护【3】。

近几年隐私保护数据发布（Privacy-Preserving Data Publishing，PPDP)【4】受到了广泛关注，隐私逐渐为人们熟知和关注。在网络时代到来之前，在个人、政府、法律、组织等的多重可信任个人或机构的保护下用户的隐私是相对安全的，自从网络出现之后，个人隐私权的相关问题逐渐扩展到了整个网络空间。主要是因为网络社会的开放特征，导致个人隐私遭受严重威胁。如何保护用户个人信息的绝对安全成为最受关注的问题。隐私保护数据发布已经成为一个新兴的、热门的研究领域【5-7】。如果因为隐私保护问题在拒绝数据的发布和共享，一方面对外发布数据，数据中含有大量有价值的信息，而数据不能发布和共享，这对数据使用者、各个行业甚至是整个社会来说都是一个巨大的损失。另一方面，若在没有考虑隐私保护的情况下，数据的发布和共享会给个体带来不可预料的精神、经济损失。通过研究数据的隐私保护发布，可以完善隐私保护在数据中的应用，使持有数据机构可以更加快速、安全地发布数据，供社会团体、研究机构研究分析，由此增加数据利用价值，并且保证了用户的隐私不被泄漏，使得数据发布和分析过程中隐私泄漏的问题得以解决。

1.2数据发布中隐私保护研究现状

1.2.1 一般性数据发布研究

目前，关于数据发布中的隐私保护研究大部分集中在结构化式数据集的数据发布上，数据以表的形式存储，行表示记录，列则表示记录的属性，每条记录对应现实中的一个个体，数据发布者对数据进行发布时，一方面要使得发布的匿名数据不泄露数据中个体的隐私信息，另一方面需要保证发布的匿名数据具有高可用性，即仍然能够根据发布的匿名数据进行较准确的数据分析，例如集合查询等。所以，数据发布中的隐私保护研究中主要解决的问题是在既满足发布数据保护个体记录的隐私安全又使得发布数据具有较高的可用性。目前，为达到隐私保护的目的已经提出了很多方法来对发布数据记录总中数据进行匿名处理。包括泛化【8】【9】、压缩【10】【11】、向原始数据中添加噪音数据【12】【13】、发布在安全范围内的边缘数据和数据交换技术等。数据发布中的隐私泄露主要可以分为身份泄露和属性泄露。当目标个体与匿名数据中的某条具体记录关联起来时就会发生身份泄露；而属性泄露则是指匿名数据会泄露目标个体的一些其他的新敏感信息。

数据发布中的隐私保护研究主要可以分为三个部分：(1)数据发布隐私保护模型的研究，它主要是作为一个衡量的标准来判断发布的匿名数据能否为包含在数据中的个体提供足够的隐私保护，例如广泛使用的 k-anonymity【6】和l − diversity【14】等隐私规则；(2)根据某种隐私数据发布模型计算匿名数据的算法研究，例如最先提出的计算满足k-anonymity 规则的匿名数据的近似算法【15】等；(3)在保证害隐私保护的前提下，提高发布数据的可用性研究，例如Anatomy[1]方法等。由P. Samarati 和L. Sweeney【6】提出的k-匿名模型（k-Anonimity），该模型对数据表中的准标识符属性进行了约束，要求发布的数据表在准标识符属性上的任意一条记录都无法与其他k-1 条记录区分，该模型虽然

可以避免身份暴露，但却常常发生属性暴露。A. Machanavajjhalla 等【14】进一步提出了l-多样化模型（l-diversity），它要求每个分组中不同的敏感属性取值不应少于l 个。LiNinghui 等【16】提出t-接近模型（t-closeness），它要求每个等价类中所有敏感值的分布要与原始数据表中敏感值的分布接近。上述模型均是先删除身份标识属性，而后对准标识属性进行适当的概化。2008 年，杨晓春等人[2]经过对多敏感属性数据的隐私保护问题深入研究后，提出了一种多维桶分组技术，该方法是一种基于有损链接的面向多敏感属性数据发布隐私保护方法。2011 年，刘善成【17】等人对多维桶分组技术进行深入分析和研究，提出了（g，l）-分组技术，不仅对分组中敏感值的多样性进行约束，还对不同敏感值之间的差异性提出了要求。

1.2.2 个性化数据发布研究

自从2006 年Xiao Xiaokui,Tao Yufei[7]等人第一次提出了个性化匿名的概念之后，个性化隐私保护技术迅速吸引了众多学者，成为数据发布中个人隐私保护技术的研究热点之一。Ye【18】等人通过对已有的（，k）-anonymity 模型进行研究和改进，提出泛化敏感属性值的方法，实现了个性化。韩建民[8]等人提出一种完全（，k）-anonymity 模型，根据不同的敏感属性值的敏感性为其指定相应的频率约束α，从而达到面向敏感值的个性化匿名的目的。程亮[23]等人提出一种满足敏感信息的多样性非相关约束的a-多样性k-匿名模型，并设计了一个改进的微聚集算法的框架代替了传统的泛化/抑制实现匿名化。黄玉蕾【19】等人提出一种基于k-匿名模型的改进算法，同时考虑不同敏感属性的整体敏感度以及用户对具体敏感属性不同需求，实现基于多敏感值的个性化隐私保护算法。

综上所述，自个性化匿名概念提出后，个性化隐私保护技术的研究大都针对单敏感属性进行的，而针对多敏感属性的个性化隐私保护的研究还很少。

1.3本文主要研究内容与组织架构

1. **数据发布中的隐私保护**

隐私保护数据发布的目的是确保个人的隐私安全，同时保证已发布数据的可用性。隐私保护数据发布场景中的4 个角色：数据提供者、数据收集者、数据发布者（如第三方发布者）、数据接收者（如研究者、数据提供者、入侵者等）。通常情况下，假设数据的发布者是值得信任的，但我们无法获知数据接收者的身份，并且数据接收者如何使用这些数据也无法明确获知，因此，我们只对数据发布过程中隐个人私保护问题进行研究。

一般待发布数据表中包含以下属性：

（1）标识符（Identifier，ID）：待发布数据表中能直接标识一条个体记录的属性，如表2-1中的Name属性。

（2）准标识符（Quasi-Identifier，QI）：准标识符是一个数据实体集的属性集合中的一组属性，通过该属性，可以将一条记录从数据表中查询出来。如表2-1 中的Age，sSx，Zipcode 属性。

（3）敏感属性（Sensitive Attributes，SA）：需要保护的信息。如表2-1中的的Disease 属性。

2.1 隐私保护中的数据泛化方法

匿名化是通过对数据表中原有的数据进行处理，以达到隐藏个体的身份或敏感信息的目的，其中处理数据的方法称为匿名化方法。现有的匿名化方法常用的主要有泛化、分解排列。例如有以下医疗数据待发布原始数据表2-1： 表2-1 原始数据表

//TODO

2.1.1 匿名化方法

（1）泛化（Generalization）

泛化(Generalization)【22】由Samarti提出，是实现匿名模型的典型方法。泛化就是将数据集中原有的精确取值转变为模糊的、范围的取值操作。其主要思想就是通过降低准标识属性值的精度，来使得数据表中在准标识属性上值相同的元组个数增加，从而降低攻击者通过准标识属性标识个体的身份或个体的敏感值的概率。准标识属性分数值型和分类型的属性，不同类型的属性泛化操作不同，数值型属性一般被泛化成区间，分类型属性则用一个更一般(相对应原属性值)的值来取代。例如：（对表2-1可进行如下泛化）年龄属

性值{10，15，20，2l，30，32}，可泛化为{[10一151，【10—15]，[20-251，[20—25]，[30—35]，

[30—35】}；邮政编码属性值{321001，321002，32103，321004，427200，427201}，泛

化为{32100\*，32100\*，32100\*，32100\*，42720\*，42720\*)。

//TODO 改成表格

经过变形的数据相对于原始数据含有较少的信息，这样既能够较好地保持数据原有的统计特性，又能保证数据的实用性。泛化是完全不显示部分（或所有）记录的一些属性值。这样会使匿名表中的信息含量降低，但是在某些情况下能够减少泛化数据的损失，达到相对较好的匿名效果。对于泛化方法，Fung等人【20】总结了五种泛化模式：全域泛化模式、子树泛化模式、兄弟泛化模式、单元泛化模式和多维的泛化。这五种泛化模式中，第一种是搜索空间最小的泛化模式，这种模式下的匿名化数据的变形度(信息损失)是最大的，第四种是搜索空间最大的泛化模式，这种模式下的匿名化数据的变形度(信息损失)是最小的。兄弟泛化模式类似于子树泛化模式，区别是可能有某些兄弟不被泛化，它产生的数据扭曲(信息损失)相比子树泛化模式要少，因为它只需要泛化那些破坏指定阈值的孩子结点。多维的泛化模式较全域和子树泛化模式产生少的数据扭曲，因为它只需要

对违背指定阈值的等价类泛化。多维的泛化对等价类中的所有记录都泛化成同样的值，但单元泛化没有这样的限制。数据泛化容易受到“维度灾难”影响。根据Aggarwal等人【21】的研究，当数据维度达到lO至15维时，泛化算法将会不可避免地丢失所有信息。为此，Xiao等人[1]提出了基于有损分解的数据匿名操作Anatomy。

（2）分解排列（Anatomy）

与泛化和隐匿相比，分解并不修改准标识属性值或敏感属性值，Anatomy等基于有损分解和置换的数据匿名操作不需要泛化层次，而可以在保留原始数据取值的前提下，保证用户的隐私不被泄露。通过将敏感属性值分组，并将敏感属性与其他属性分开发布的方式，来扰乱准标识符与敏感属性之间的关联。如图2-1所示：

图2-1 通过有损分解实现数据匿名

//TODO 画图

原本准标识符qid与敏感属性sa之间是意义对应关系（a），但是通过有损分解后，这种对应关系通过分组gid联系起来（b），原本一对一的关系被分割成通过gid维持的一对多的关系（c）。重构经过Anatomy处理的数据会出现很多不再原始待发布数据表中的记录，这种出现重构错误可以保证攻击者无法精确确定用户记录的敏感信息，达到隐私保护的目的。分解的方法主要用于实现敏感属

性多样性模型。排列的方法与分解的处理方式相像，它通过将待发布数据记录划分为若干个分组，然后在各分组中随机打乱敏感属性值的顺序，从而达到扰乱准标识符与数值型敏感属性的对应关系，这种方式主要是针对敏感属性为数值型的待发布数据集。

2.2 隐私保护中的匿名模型

2.2.1 k-匿名模型

2.2.2 L-diversity型模型

2.2.3 t-closeness模型

2.3 隐私保护原则与信息度量

2.4 本章小结

1. **多维敏感属性数据发布中的隐私泄露**

3.1 多敏感属性数据发布中的问题研究分析

在对数据发布中的隐私保护技术研究初期，大多数敏感数据发布方法都是针对单一敏感属性的保护。但是，在实际的应用中，发布的数据大多数都会涉及到多个敏感属性，特别是这些敏感属性在某些情况下会存在一些关联关系，一些属性虽然对于发布个体不是直接的敏感属性，但是这些属性却和个体的敏感属性有着明显的特定关系，所以这样的属性也应该归类到个体的敏感属性被保护。例如表3-1-1 为将要发布的原始医疗信息，从表中可以看出，敏感属性主治医(Physician) 和疾病(Disease)之间存在着关联性，通过背景知识我们也可以知晓，一个主治医生专治哪几种疾病，其具体的关联性如表3-1-2 所示。

表3-1-1 原始医疗信息表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tuple ID | Name | Age | Sex | Zipcode | Physician | Disease |
| t1 | Sam | 23 | M | 821071 | John | Flu |
| t2 | Anne | 44 | F | 821023 | John | Pneumonia |
| t3 | Mike | 56 | F | 821045 | John | Cancer |
| t4 | Lily | 35 | M | 821123 | Bob | Flu |
| t5 | Harry | 25 | F | 821031 | Bob | Pneumonia |
| t6 | Mona | 30 9 | M | 821035 | Anne | Gastritis |
| t7 | Tony | 40 | F | 821110 | Anne | Gastritis |
| t8 | Lucy | 37 | M | 821115 | Hugo | HIV |
| t9 | Tim | 60 | M | 821134 | Marry | Flu |

3-1-2 属性间关联表

|  |  |
| --- | --- |
| Physician | Disease |
| John | Flu,Pneumonia,Cancer |
| Bob | Flu,Pneumonia |
| Anne | Gastritis |
| Hugo | HIV |
| Marry | Flu |

现在假如我们需要对3-1-1表数据内容进行数据发布，并以l-diversity 匿名算法为例进行。我们对表3-1-1原始医疗信息进行匿名化处理，并通过分组算法得到数据发布表，结果如表3-1-3 所示。由表3-1-3 可以看出，由于医院中医生的主治哪些疾病是可以很容易获取的，也就是攻击者能够很容易获得表3-1-2内容的背景知识，若攻击者得知该个体的Physician 属性值为“John”再联合攻击者掌握的个体的准标识属性确定个体属于Group ID为3的分组。此时，攻击者推测出该个体的Disease 属性值的概率将高于1/3，这就违反了L多样多样性原则，造成隐私泄露的风险升高。这正是由于敏感属性Physician和Disease 间存在着关联性，即使数据发布表中的敏感属性满足l-diversity 匿名模型，依然存在隐私泄露风险。因此针对单敏感属性的隐私保护技术并不能直接用于多敏感属性数据发布，否则会给个体隐私数据的保护带来很大的挑战，所以对于多敏感属性的数据发布仍存在隐私泄漏风险。为了适应实际应用中的数据发布，面向多敏感隐私属性的数据发布方法的研究应受到重视。

表3-1-3 医疗发布数据3-diversity匿名表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tuple ID | Age | Sex | Zipcode | Physician | Disease |
| t1 | [20,30] | M | 8210\*\* | John | Flu |
| t5 | [20,30] | F | 8210\*\* | Bob | Pneumonia |
| t6 | [20,30] | M | 8210\*\* | Anne | Gastritis |
| t8 | [31.40] | M | 821\*\*\* | Hugo | HIV |
| t4 | [31,40] | M | 821\*\*\* | Bob | Flu |
| t7 | [31,40] | F | 821\*\*\* | Anne | Gastritis |
| t3 | [41,] | F | 821\*\*\* | John | Cancer |
| t2 | [41,] | F | 821\*\*\* | John | Pneumonia |
| t9 | [41,] | M | 821\*\*\* | Marry | Flu |

3.2 多敏感隐私属性的数据发布方法

因为多敏感属性数据隐私保护有一些特殊的要求，为了防止由于缺少整体性而造成失去隐私敏感属性的连锁攻击，许多学者提出了针对多敏感属性数据的隐私保护模型和方法。文献[3]提出了多敏感属性l-diversity 概念，并对其进行如下定义：

多敏感属性l-diversity. 设数据表T 中有若干个准标识符属性QI 和敏感属性SA，从QI 中任意选取一个属性将其作为唯一的敏感属性，其余的准标识符属性和敏感属性均作为准标识符属性，则此时若数据表T 均满足l-diversity，则说明数据表T 满足多敏感属性l-diversity。

由以上定义可知，多敏感属性l-diversity 规则要求每个敏感属性上的每一个敏感值与所有其他敏感属性上对应的敏感值的个数不少于l 个[12]，这一概念可以很好的解决多敏感属性数据发布的隐私保护问题。但是，当敏感属性个数增加时，每个等价类为了满足多敏感属性l-diversity 规则就必须包含更多的记录，这必然会导致数据表泛化程度加剧，从而造成大量的信息损失。

3.2.1基于多维桶分组技术的隐私数据发布方法

2008 年，杨晓春等人[2] 首次提出了以l-diversity 模型为基础的多维桶分组技术来解决多敏感属性数据发布的隐私保护问题。

多维桶分组技术的基本思路是，首先，将多个敏感属性看成一个高维复合敏感属性向量，也就是说，一个敏感属性对应一维；其次，使用多维桶的向量模型，将数据表中的记录映射到多维桶上；最后，按照某种方法在构造的多维桶上进行分组操作，使分组中的记录尽可能是在各维度上取值都不相同的桶中提取出来的。

多维桶分组技术重点是将多个敏感属性作为高维复合敏感属性来构造桶，并提出以下定义：

假设用户待发布的数据表为T{A1，A2，A3，···，Ap，S1，S2，···，Sd}，其中Ai（1ip）是待发布数据T的准标识属性（QI），p代表准标识属性的个数。Sj（1jd）是待发布数据T的敏感属性，d代表敏感属性的个数。设T中记录个数为n，即|T|=n，那么发布数据表中每条记录记为ti（1in），另t[X]标识记录t在X属性上的取值，其中X{A1，A2，A3，···，Ap，S1，S2，···Sd}。

（1）复合敏感属性.。待发布数据表T中所有的敏感属性构成一个复合敏感属性，记作S。其中第i个敏感属性可看作复合敏感属性的第i 维，记为Si， Dom（Si）为Si 的取值范围，指该敏感属性的所有取值，|Si|为D（Si）的基数，指该敏感属性取值的个数。

（2）复合敏感属性向量[2]. 待发布数据表T中任意记录t的全部敏感属性取值构造成向量模式<t[s1]，t[s2]，…，t[sd]>，称这样的向量模式为复合敏感属性向量。

（3）分组[4]. 一个分组是T 中记录的子集。T 中每一个记录属于且仅属于一个分组，T中所有记录的分组记为GT{G1，G2，G3，···，Gm}，其中m为最终分组数，并且 =T，并且=（1ijm）。

（4）单敏感属性L-多样性[5]。对于一个分组G，G中只包含单敏感属性的记录，假设v为G中出现频度最大的敏感属性取值，且c(v)为v在G中出现的次数，如果满足，|G|为G 中记录的个数。

（5）复合敏感属性L-多样性分组. 对于一个包含复合敏感属性的分组G，如果G中的任一一维敏感属性Si（1id）都满足单敏感属性L-多样性，则该分组满足复合敏感属性满足l-多样性。那么对于T中所有分组GT{G1，G2，G3，···，Gm}，如果其中每个分组Gi（1im）都满足复合敏感属性L-多样性性质，则称GT为T上的复合敏感属性L-多样性分组。

有了以上定义以后，我们知道待发布数据表T若存在多个敏感属性，仅仅满足单敏感属性的L-多样性原则无法保证所有敏感属性的隐私保护需求，针对多敏感属性的数发布，需要满足多敏感属性中每个属性都满足L-多样性原则，即需要得到待发布数据表T的复合敏感属性L-多样性分组。杨晓春等人提出基于有损连接技术的支持多敏感属性的隐私数据发布多维桶分组技术，目标是找到具有多敏感属性的待发布数据表T的分组方案，使得到的分组均满足复合敏感属性L-多样性。

多维桶的构造原理如下：复合敏感属性的每个维度对应多维桶的一维，将数据表T中的数据记录根据其复合敏感属性向量每一维的值分别映射到相应的桶中。设数据表T 的敏感属性个数为d，构造的d 维桶记为BUK（S1，S2，……，Sd），其中每个桶记为buk（s1，s2，……，sd），每个桶的大小记为size（buk（s1，s2，……，sd）），即含的记录数。以表 3-1-1 的原始数据为例，建立多维桶如表3-1-4 所示。

表 3-2-1 由表3-1-1构造的d维桶（d=2）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Flu | Pneumonia | HIV | Gastritis | Cancer |
| John | {t1} | {t2} |  |  | {t3} |
| Bob | {t4} | {t5} |  |  |  |
| Anne |  |  |  | {t6,t7} |  |
| Hugo |  |  | {t8} |  |  |
| Marry | {t9} |  |  |  |  |

在得到表3-1-4的多维桶后，分别采取MSB的贪心算法，包括最大桶优先算法，最大单维桶优先算法和最大多维桶优先算法来得到最后的满足复合敏感属性L-多样性分组的发布数据T’。

基于多维桶的分组算法为复合敏感数据发布的隐私保护提供了分组方案，并解决了多敏感属性下的隐私泄露问题，但仍然存在一些不足之处：

（1）多维桶分组技术在实现分组算法的时候，只考虑了最后分组对原数据的覆盖率问题，以贪心策略尽可能的得到更多的满足复合敏感属性L-多样性的分组，没有考虑到在实际数据发布中，每一维敏感属性的取值中，可能存在敏感度高低的问题，例如医疗数据发布表当中，个体敏感属性Disease取值中，HIV的敏感度肯定高于Flu。这就导致，多维桶分组算法在得到待发布数据表的所有分组后，可能会存在某个分组中，某个敏感属性的取值均是敏感属性度较高的情况，出现敏感信息分布切斜的现象，从而导致个体的敏感信息泄露。

（2）多维桶分组技术考虑到实际应用中某些元组包含着更重要的信息，需要尽量保留在发布的数据中，从而提出加权多维桶分组技术。加权多维桶分组技术只是考虑到部分重要数据的权值，在分组时先将权值高的记录先加入分组，这种方式虽然达到了保留重要数据的目的，但是由于重要数据一般都是敏感属性较高的数据，这种加权分组方式在实际应用中很有可能加剧敏感信息倾斜，造成个体隐私泄露。为解决这一问题，我们应该在保留高权值敏感信息的情况下同时保证高敏感度信息在分组中分布的均匀性。

3.2.2 基于L-覆盖性聚类分组的隐私数据发布方法

金华等人分析多维桶分组算法存在分组效率低，可能会存在由于每次分组选取桶的顺序问题造成大量不必要的数据记录遭到隐匿的问题，进一步对多敏感属性的医疗数据发布方法进行研究，提出了基于有损连接技术和相同敏感属性集的L-覆盖性聚类分组算法[6]。在多维桶提出的多敏感属性L-多样性的基础上给出以下定义：

（1）移除。对于一个分组G，若移除G中任意一条记录中的某一敏感属性值t[Si]（tG，1id）,需要将G中所有包换t[Si]的记录都删除。

（2）多敏感属性L-覆盖性。在一个分组G中，若至少需要移除L个敏感属性值，才能将G中所有记录移除，则称分组G满足多敏感属性L-覆盖性。

（3）相同敏感属性集。对于待发布数据表T中，包含同一敏感属性取值的所有记录组成的集合称为相同敏感属性集，记为SID（v）。待发布数据表T中，记录t的所有敏感属性值t[Si]的相同敏感属性集的并集称为记录相同敏感属性集，记为t.TSID，t.TSID=(t[Si])。对于一分组，中所有记录的记录相同敏感属性集的并集称为分组相同敏感属性集，记为.GSID,.GSID=。

（4）平均概率泄露度。由于需要处理完所有剩余记录，数据集T中肯定会存在仅仅满足多敏感属性L-覆盖性而不满足复合敏感属性L-多样性的分组。设得到的发布数据分组为GT{G1，G2，G3，···，Gm},||L，G中所有敏感属性的不同取值个数为n，每个敏感属性的取值为（1in）,在的出现的频率记为，则分组的概率泄露度定义为,则数据集的平均概率泄露度为=，其中|T|标识待发布数据集T中记录总数。

L-覆盖性聚类分组方法的主要思想是通过聚类的方法将满足L-覆盖性的记录进行分组，采用聚类思想，首先在数据集中顺序选取L个满足L-覆盖性的记录构成分组。然后对剩余记录分两步处理：第一步处理剩余记录中可以添加到其他分组而且仍满足多敏感属性L-多样性性质的记录；第二步是处理第一步剩余的记录，将剩余记录均匀地添加到分组较小的分组中，以降低平均概率泄露度。我们以表3-1-1为原始数据表为例，利用L-覆盖性聚类分组算法进行分组。另L=3，首先将数据集T中的每一条记录看做是一个分组，并计算他们各自的GSID。根据GSID的定义我们可以得到G1.GSID={t1,t2,t3,t4}，G2.GSID={t1,t2,t3,t5,t9}，G3.GSID={t1,t2,t3}，G4.GSID={t1,t4,t5,t9}，G5.GSID={t2,t4,t5}，G6.GSID={t6,t7}，G6.GSID={t6,t7}，G7.GSID={t6,t7}，G8.GSID={t8}，G9.GSID={t1,t4,t9}。按照L-覆盖性聚类算法顺序选取规则，首先选取G1，然后选择不在G1.GSID中含有的记录分组G6，得到G1,6.GSID={t1,t2,t3,t4,t6,t7}，然后选取不在G1,6中含有的记录分组G8，此时已得到一个分组{t1,t6,t8}，然后移除记录t1，t6，t8。算法依次循环进行得到另一个分组{t2,t7,t9}，剩余记录{t3,t4,t5}无法满足L-覆盖性（L=3），t5能够加入到第一个分组中得到{t1,t5,t6,t8}满足复合敏感属性L-多样性（L=3），

t3与t4加入到第二个分组得到{t2,t3,t4,t7,t9}，满足L-覆盖性（L=3）,最终得到的发布结果如表3-2-2所示。

表 3-2-2 L-覆盖性积累分组算法发布数据结果

|  |  |  |
| --- | --- | --- |
| Tuple ID | QIs | Group ID |
| t1 | ··· | G1 |
| t2 | ··· | G2 |
| t3 | ··· | G2 |
| t4 | ··· | G2 |
| t5 | ··· | G1 |
| t6 | ··· | G1 |
| t7 | ··· | G2 |
| t8 | ··· | G1 |
| t9 | ··· | G2 |

|  |  |  |
| --- | --- | --- |
| Group ID | Physician | Disease |
| G1 | John | Flu |
| Bob | Pneumonia |
| Anne | Gastritis |
| Hugo | HIV |
| G2 | John | Pneumonia |
| John | Cancer |
| Bob | Flu |
| Anne | Gastritis |
| Marry | Flu |

L-覆盖性分组算法解决了多敏感属分组后存在的剩余记录的问题，并提出了隐私平均概率泄露度的概念，但是含有敏感属性的数据发布中，最重要的就是保护个体记录的隐私不被泄露，所以该算法的不足之处也显而易见：

（1）满足L-覆盖性的复合敏感属性分组，并不一定满足复合敏感属性L-多样性，特别是在剩余记录存在大量相似敏感属性的时候，由于L-覆盖性算法在分组完成后为了将剩余记录全部加入已分组当中，会造成部分分组中敏感属性相同的记录出现的概率大与1/L。

（2）L-覆盖性分组算法总是顺序选取记录来进行分组，而且没有考虑敏感属性的度量问题，该算法虽然解决了剩余记录问题，但却在个性化分组和分组效率上存在比较大的局限性，分组效果通常不理想，在剩余记录较多的情况下，容易造成个体记录的隐私泄露。

3.3 多敏感属性的个性化隐私保护

通常情况下，对于待发布数据表T，都是由数据发布者来决定数据发布表中的敏感属性信息，通过设定一些约束参数来对待发布数据进行隐私保护。自从Xiao等人在2006年首次提出针对数据发布中的个性化匿名发布概念以后，个性化隐私保护就成为了数据发布中隐私保护研究的重要研究方向。

个性化隐私保护是指在进行隐私保护时，由数据发布者决定待发布数据表中的隐私属性，针对不同场景和发布数据指定不同的隐私保护策略与个体隐私保护的强度，从而满足不同的人对不同敏感属性的不同约束要求，在保护个体隐私的前提下，达到数据个性化发布的目的。所以个性化隐私保护可以很好的满足不同场景下对于不同隐私得保护的要求，并能够在一定程度上克服全局准标识属性匿名化编码造成的对敏感属性的保护“不足”和“过度”保护等问题。个性化数据法布中隐私保护目前的研究主要分为两个反面，一类是面向数据发布表中每条个体记录的个性化隐私保护；另一类则是针对数据发布表中所有敏感属性取值的个性化隐私保护。

1. 面向个体记录的个性化隐私保护方法

面向个体记录的个性化隐私保护策略的主要研究对象是针对待发布数据表中的每条个体记录，即研究的对象是个体。在数据发布者处理待发布数据时，为满足个性化数据发布的需求，需要从每个个体的实际隐私保护需求出发，对每条个体记录制定不同的个性化约束，对个体和与其相关的敏感属性之间的关联性进行一定的约束和限制。

面向个体记录的个性化隐私保护方法从本质上看来能够很好的制定数据发布表的个性化发布方案，但实际应用中待发布的数据集一般比较大，若要为数据集中每条记录都设定不同的个性化约束就需要非常大的任务量，因此，这种面向个体记录的单独设定个性化约束的数据发布方式虽然达到了良好的个性化需求，但在实际操作时缺乏可行性，存在一定的局限性。

（2）面向敏感属性值的个性化隐私保护方法

面向敏感属性值的个性化隐私保护方法主要是针对待发布数据表中的敏感属性的所有取值，以发布数据的敏感属性值为基础，数据发布者可以根据数据表中的敏感隐私信息不同的敏感值设定不同的个性化约束，实现个性化隐私保护。

面向敏感值的个性化隐私保护方法与面向个体记录的个性化隐私保护方法具有更高的可行性，可以很好的解决数据发布中隐私保护技术对隐私信息的过度保护从而造成发布数据的可用性降低或对隐私信息的保护不足而造成隐私泄露的问题。目前针对对于个性化隐私保护的技术的研究大都针对于单个隐私属性的数据发布，在待发布数据存在多敏感属性的情况下，单敏感属性的个性化隐私保护方法并不适用，因此针对多敏感属性的个性化隐私保护数据发布方法还需要更加深入的研究。

3.3.1 面向多敏感属性的个性化数据发布算法

（1）完全（a,k）-anonymity模型

韩建明等人[8]根据简单（a,k）-匿名模型和一般（a,k）-匿名模型提出完全（a,k）-匿名模型，该模型主要思想是根据不同敏感属性质的敏感度不同设置不同的频率约束a，以此实现对不同敏感属性的值在同意等价类分组中出现的频率进行控制，实现针对敏感属性值的个性化分组，达到个性化发布的目的。

简单（a,k）-匿名约束是面向一个特定的敏感值的，给定一个待发布数据表T，经过匿名分组后得到待发布数据表T’，对匿名表中任意一等价类分组G，给定一个敏感属性值v，（G，v）为等价类分组G中包含敏感属性值v的元组集合，如果v在等价类分组G中出现的频率都不大于a，即，则称敏感属性值v满足简单（a，k）匿名模型。一般（a，k）-匿名模型是将简单（a，k）-匿名模型从单个敏感属性值的约束扩展到对所有敏感属性值的约束，不仅仅限制单个的敏感属性值在任意等价类中出现的频率小于a，而是限定所有的敏感属性取值在任意等价类分组中出现的频率均小于a。

完全（a，k）-匿名模型是在简单（a，k）-匿名模型和一般（a，k）-匿名模型的基础上进行的推广。完全（a，k）-匿名模型针对待发布数据表T中的每一敏感属性v设定相应的频率约束av（0av1），要求得到的发布数据表T’中任意等价类分组中个敏感属性值v均满足（av，k）-匿名模型的约束。这里根据实际情况，若敏感属性值v的敏感属性越强，为了增大对其隐私保护力度，则相应的av就应该越小；敏感属性值的敏感度越弱，则对应的av就越大。如在医疗数据发布表中，敏感属性“疾病（Disease）”的取值“HIV”与“Flu”，明显“HIV”的敏感度强于“Flu”的敏感度，则对应的“HIV”的频率约束值aHIV应该小于“Flu”的频率约束值aFlu。例如表3-3-1的匿名数据发布表满足完全（a，k）-匿名模型，其中aHIV=0.4，aFlu=0.6，aPneumonia=0.4，aGastritis=0.4。

表 3-3-1 完全（av，3）-匿名表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tuple ID | QIs | Zipcode | Physician | Disease | Group ID |
| t1 | ··· | 8210\*\* | John | Flu | G1 |
| t5 | ··· | 8210\*\* | Bob | Pneumonia |
| t6 | ··· | 8210\*\* | Anne | Gastritis |
| t8 | ··· | 821\*\*\* | Hugo | HIV | G2 |
| t4 | ··· | 821\*\*\* | Bob | Flu |
| t7 | ··· | 821\*\*\* | Anne | Gastritis |
| t9 | ··· | 821\*\*\* | Marry | Flu |

在完全（a，k）-匿名模型中关于as的设定原则是，设定值as应该不小于敏感值s在原始待发布数据表中的出现的频率，否则难以生成满足完全（a，k）-匿名约束的发布匿名表。设待发布的数据表为T，|T|为数据表中元组个数，G为发布数据表中的等价类分组，S为敏感属性，vs为敏感属性的取值， avs为敏感属性vs的频率约束，则avs应该满足如下关系式：

1. 基于最小选择度优先的多敏感属性分组算法

杨静等人[9]在研究多敏感属性的隐私保护问题的时候，在传统单敏感属性L-多样性的基础上，利用拓扑空间中覆盖的思想定义了多敏感属性的L-多样性原则，引入了基于值域等级划分的个性化隐私保护方案，针对多敏感属性隐私保护提出了一种基于最小选择度优先的分组算法，在满足多敏感属性L-多样性原则的同时，实现敏感属性的个性化隐私保护需求。

该算法首先将敏感属性进行值域划分。将给定的敏感属性S，按照敏感属性S中的不同敏感属性取值的敏感度由高到低进行排序，然后对敏感属性取值划分成m个等级，记为CG（S）={LS1，LS2，LS3，···，LSm}。若CG（S）满足以下关系：=Dom（S）且SiSj=（1ijm），则就称CG（S）为S的一个值域等级划分。且SDegree（LSi）表示敏感属性S的在等级LSi的所取敏感度。所有敏感属性的值域等级划分完成后，发布数据表中每条个体记录的每个敏感属性都存在且只存在一个值域等级中。每条个体记录的记录敏感度为每个敏感属性的敏感度之和，即：

TDegree（t）=（t[Si]）

最小选择度优先的分组算法是根据个体记录的选择度执行的，在待发布数据表中，每条个体记录的选择度Select（tj）为tj中每一个敏感属性值v在待发布数据表T中出现的频率之和，考虑到个性化需求，所以单个个体记录的个性化选择度为PSelect（tj）表示为：

Select（tj）(tj)

其中为敏感属性在待发布数据表中出现的频率。（tj）是待发布数据表中个体记录tj中相异敏感属性值的集合。

最小选择度优先算法是一种启发式方法，基本策略是首先选择个性化选择度最小的元组作为等价类分组的初始元组，然后将具有不同敏感属性值的其他元组加入到当前等价类分组中，如果该等价类元组的元组数L，则将该等价类分组并入到待发布分组中，否则并入到待处理元组集中。循环上述步骤直至处理完待发布数据集中所有元组，从而得到最后满足多敏感属性隐私保护策略的发布数据所有分组。

完全（a，k）-匿名模型与基于最小选择度的数据发布算法都是针对多敏感属性数据发布中个性化发布的方法，完全（a，k）-匿名模型中虽然考虑了单个敏感属性的敏感度取值，但是对于含有高敏感度取值的元组没有进行个性化分组，可能造成含有高敏感度的元组（通常是比较总要的元组）被隐匿，造成整体数据集的可用性降低。基于最小选择度优先的启发式算法，结合了敏感属性值的敏感度和待发布数据的个体记录进行选择度量从而制定个性化分组，比较好的保留了选择度低的元组（敏感度高的元组），但算法只考虑了敏感属性值的敏感度而没有考虑敏感属性本身的敏感度问题。另外，最小选择度算法首先保留敏感度高的元组划入分组方法，有可能造成敏感度高的元组划分到同一等价类分组中，造成隐私属性倾斜，容易受到同质攻击。

3.4 本章小结

本章主要是针对多敏感属性数据发布中存在的问题进行了细致的分析，并从常规的多敏感属性数据发布和个性化多敏感数据发布两个方面展开讨论，讨论了常规多敏感数据发布中基于多维桶和L-覆盖性两种算法，个性化发布中的基于完全（a，k）-匿名模型的算法和基于最小选择度优先的数据发布方法，并分析各种算法解决的主要问题并分析这些算法在数据发布中存在的不足之处，引出下章本文提出的面向多敏感数据的发布算法以及个性化发布模型。

1. **面向多维敏感属性的数据发布**

4.1 多敏感属性隐私数据发布问题

敏感数据发布与共享环境中的个体隐私信息的安全问题一直是数据隐私研究的热点。在关系型数据发布中，Sweeney等人最早提出的k-匿名模型可保护但敏感属性下的隐私数据发布不受链接攻击[12,13]。文献[5]在分析k-匿名模型在某些情况下并不能保证隐私信息的安全。例如，在对数据表中准标识属性值概化后，具有相同准标识属性取值的大部分或者左右敏感属性的取值相同，那么攻击者只要确定个体记录属于哪个分组就能高概率地推断出个体的隐私信息，因此文献[5]提出了L-多样性概念，对匿名化数据表中，单个等价类分组中出现频率最高的敏感属性值的个数要求不大于1/L。

对于多敏感属性数据发布的研究，杨晓春等人提出的基于多维桶分组技术的方法对待发布数据表进行分组，是得到的分组在复合敏感属性的前提下满足L-多样性模型，达到隐私保护的要求。该方法较好地实现了对多敏感属性数据发布的隐私保护，但该分组方法效率较低，往往会因为选取桶的顺序问题造成分组效果不太理想，数据隐匿率较高，降低了发布数据的可用性。文献[6]提出的基于L-覆盖性分组算法，虽然解决了数据隐匿率等问题，但得到的等价类分组并不一定满足复合敏感属性L-多样性，特别是在剩余记录存在大量相似敏感属性的时候，由于L-覆盖性算法在分组完成后为了将剩余记录全部加入已分组当中，会造成部分分组中敏感属性相同的记录出现的概率大与1/L，存在隐私保护力度不足的问题。

因此，为了解决多敏感属性分组中分组效率不高与隐私保护力不足造成隐私泄露的问题，本章提出一种新的基于类二部图匹配的分组算法，对具有多维敏感属性的原始数据进行分组，使得各分组满足L-多样性，且降低数据隐匿率。

4.1.1 发布数据表中的属性定义

根据现有匿名研究[10,11]，待发布数据表T中的属性可以分为以下四类：

（1）标识符（Identifier，ID），能唯一标识数据表中个体记录或者机构具体身份的属性，例如身份证号码，手机号码，社会保险号等。

（2）准标识符（Quasi-Identifier，QID），数据表中联合起来能够标识个体记录的属性，如性别，出生日期，邮政编码，年龄等。

（3）敏感属性（Sensitive Attribute，SA），数据表中包含个体敏感信息的属性，如薪资，患病记录，职业和位置等。

（4） 非敏感属性（Non-Sensitive Attribute，NSA）数据表中除了以上三种属性之外的属性。

4.1.2 有损连接发布

Anatomy等在基于有损分解发布数据表的方式不需要对发布数据表进行泛化匿名操作，可以在保留原始数据表属性取值的前提下，保证个体记录的隐私信息不被泄露，通过分解准标识属性（QID）和隐私属性（SA）之间的对应关系，有损连接发布方式可以在直接发布原始数据表中的原始准标识属性值和敏感属性值的情况下保证发布数据满足L-多样性原则，保护用户隐私。

表 4-1-1 原始表

|  |  |  |
| --- | --- | --- |
| Tuple ID | QIDs | Sensitive Attribute |
| t1 | qid1 | sa1 |
| t2 | qid2 | sa2 |
| t3 | qid3 | sa3 |
| ··· | ·· | ··· |

以上我们有待发布数据表 4-1-1，假设元组{t1，t2，t3}为一个等价类分组，我们可以看到元组中的准标识属性值{qid1，qid2，qid3}和敏感属性{sa1­，sa2，sa3}是一一对应关系，经过有损分解，将待发布表分为准标识属性表和敏感属性表分成连个表发布，两个表之间仅通过分组编号相连，如表4-1-2所示。现在原有的一对一关系被分割成通过Group ID维持的一对多关系，攻击者无法保证通过准标识符唯一确定其敏感属性，若敏感属性Sensitive Attribute满足L-多样性，则真个数据发布表满足L-多样性原则。

表 4-1-2 有损分解发布表

|  |  |  |
| --- | --- | --- |
| Tuple ID | QIDs | Group ID |
| t1 | qid1 | G1 |
| t2 | qid2 |
| t3 | qid3 |
| ··· | ·· | ··· |

|  |  |
| --- | --- |
| Group ID | Sensitive Attribute |
| G1 | sa1 |
| sa2 |
| sa3 |
| ··· | ··· |

4.2 基于类二部图边选择的多敏感数据分组--BES

设待发布关系型数据表T={A1，A2，A3，···，Ap ，S1 S2 S3，···，Sd}，其中Ai{A1，A2，···，Ap}（1ip）为准标识属性，Sj{S1，S2，···，Sd}（1jd）为敏感属性。待发布数据表T中共有n条记录，即|T|=n，数据表中每条个体记录称为一个元组，元组标识为ti（1in）。令t[X]标识数据表中元组t在X属性上的取值。

定义4.1 分组[5]。将待发布数据表T中所有元组分为若干组，记为GT，GT={G1，G2，···，Gm}，并且 =T，并且=（1ijm）。则称GT为待发布数据T的分组。

定义4.2 复合敏感属性[2]。待发部数据表中的所有敏感属性组成复合敏感属性S={S1，S2···，Sd}，S为复合敏感属性集合，则|S|2。SiS（1id）标识数据表中的第i个敏感属性，Dom（Si）表示敏感属性Si的值域，|Si|标识Dom（Si）的基数，即Si所有可能取值的个数。

定义4.3 单敏感属性L-多样性。对于单敏感属性下得到的一个分组G，设SSet（G）为分组G中所有不重复的敏感属性取值。vi（i）为分组G中某一敏感值，viSSet(G), 标识vi出现的次数，若分组G中满足： ，则称该分组满足单敏感属性L-多样性。

定义 4.4 多敏感属性L-多样性。若一个包含多敏感属性分组G，如果其中每个元组的每一条记录上的敏感属性的取值都满足单敏感属性L-多样性，则称分组G满足多敏感属性L-多样性。

根据以上定义，针对一般多敏感属性隐私数据发布的问题主要是将待发布数据表T进行分组得到发布表T’，T’中所有分组是满足隐私保护要求的，本文要求是得到的分组均满足多敏感属性L-多样性，通过有损分解发布数据表T’,以下提出一种新的基于二部图边选择的分组算法（Bigraph-similar Edges Selection），得到多敏感属性L-多样性分组，发布数据表满足多敏感属性L-多样性模型。

4.2.1 BES基本思想

本章提出的基于类二部图的边选择的分组技术（Bigraph-similar Edges Selection）是为解决多敏感隐私属性数据发布中的分组问题，目的是找到多敏感属性待发布数据表T上的分组方案，得到分组GT，使得每个分组均满足多敏感属性L-多样性。因此，BES分组方法首先需要将多敏感属性数据表T中所有元组映射到二部图上。

定义4.5 二部图。无向图BG = <V,E>的结点集V能够划分为两个子集V1,V2，满足V1∩V2 =，且V1∪V2 = V（全集），使得BG中任意一条边的两个端点，一个属于V1，另一个属于V2，则称G为二部图或二分图（Bigraph）。

针对待发布医疗数据表3-1-1中，共有两个敏感属性Physician与Disease，取值集合分别为：

SPhysician={John，Bob，Anne，Hugo，Marry}

SDisease={Flu，Pneumonia，Cancer，Gastritis，HIV}

其中SPhysician∩SDisease =,则我们可以将每一维敏感属性作为图的一个点集，每个敏感属性的取值为图上一点，根据表3-1-1的各元组可得到如下图的二部图：

图 4-2-1 表3-1-1数据表敏感属映射的类二部图

//TODO

图中每条边即代表从待发布数据表T中的一个元组中的敏感属性取值，如若待发布数据表中的敏感属性超过2个，例如在表3-1-1的中添加第三个敏感属性Marital-status，敏感属性取值SMarital-status={Divorced，Separated，Never-married，Married}。则可根据表数据构造出图4-2-1类二部图（Bigraph-similar）：

图 4-2-2 3个敏感属性映射的类二部图

定义4.6 元组边（t.E）。发布数据表T中ti(1in),tiT。由ti映射到类二部图的所有边称为该元组边的元组边，记为ti.E。V(ti.E)标识该元组边经过的所有点的集合。所有元组边构成的集合称为类二部图的元组边集TE=。例如图4-2-1中，元组t1的元组边为t1.E={(John,Flu)},t1元组边经过的点集V（t1.E）={John，Flu}。

根据待发布数据表T映射得到的类二部图表示为BG，BG=<VS,TE>，其中VS={VS1, VS2, ···, VSd}，其中VSi（1id）表示敏感属性Si的所有属性值映射到类二部图中的点的集合。例如图4-2-1中，敏感属性“Disease”对应的点集合VDisease={ Flu，Pneumonia，Cancer，Gastritis，HIV}。在得到所有元组映射的类二部图后，基于边选择的方法采用某种策略尽可能选择多的元组边作为一个分组，且这些元组边的敏感属性点集Vti没有交集。如上文所述，BES方法在发布数据时采用分组内有损连接的方式发布数据，因此在不破坏L-多样性的前提下，分组越小造成的数据利用度越高，信息损失也越小。理想情况下若得到的分组大小为L，并满足多敏感属性L-多样性，就要求每个分组中的每条个体记录的敏感属性值值出现一次。处理剩余记录时，在满足多敏感属性L-多样性的前提下，可能某些分组的记录条数会超过L，会造成附加的有损连接信息损失。这里采用文献[2]定义的附加信息损失度=，其中m为得到的分组数。

定义4.7 不相交边选择。在类二部图BG=<VS,TE>中，在为当前分组G选取一条元组边t.E时，若该元组边与已加入当前分组的所有元组边均不相交,即

，则称为不相交边选择。

基于类二部图边选择的分组方法是一种以固定分组大小，采用贪心策略在类二部图上依次对元组边作不相交边选择，选取L个元组边对应的元组构成分组，重复进行，尽可能多地得到大小为L的分组，最后在不破坏多敏感属性L-多样性的前提下，将剩余元组加入到已有分组中。最后将不包含在任何分组的个体记录从发布的数据中隐匿掉。采用数据隐匿率（Suppress ratio）来衡量分组后隐匿的数据记录占发布数据表中的比例。数据隐匿率定义，Ns为隐匿的个体记录数，由上可知道SuppRatio越小，隐私数据越少，理想情况下，SuppRatio=0。本文将数据隐匿率和附加信息损失度一起作为发布数据的衡量标准。

4.2.2算法描述

BES分组算法主要分为3个步骤：

（1）将待发布数据表的多个敏感属性提取出来，构建类二部图并得到所有的元组边。

（2）分组阶段在已有的类二部图中依次做不相交边选择，得到每一个敏感属性取值都互不相同的L个元组边构成一个有效分组。循环进行，直到剩余的元组边中无法再做不相交边选择操作。

（3）剩余记录处理阶段。将第（2）步中剩余的元组依次遍历，在不破坏原有分组满足多敏感属性L-多样性的前提下将记录添加到分组中，最终将不属于任何分组的个体记录隐匿。

算法. 基于类二部图的边选择分组算法

输入：待发布数据表T{A1，A2，A3，···，Ap，S1，S2，···，Sd}，多样性参数L

输出：准标识属性表QIT，敏感属性表ST

步骤：

1. 提取T中敏感属性值，构建类二部图BG，得到所有记录的元组边TE
2. while TE中的元组边不为空
3. 遍历TE-> t[i···n].E
4. if ti.E能在当前分组G上作不相交边选择
5. 将ti将入到当前分组G
6. if 当前分组G中元组数等于L
7. 将当前分组G加入到已完成分组集合GS中，并将G中所有

元组对应的元组边从TE中移除

1. 停止遍历，重新进入到while循环
2. if 当前分组G中元组数不足L
3. 将分组G中所有元组对应元组放入待处理元组集合RT，并将其

对应的元组边从TE中移除

1. 遍历RT->t[i···|RT|]
2. for G[j…|GS|]
3. if ti加入到GSj中GSj依然满足多敏感属性L-多样性
4. 元组ti加入到GSj中，并将ti从RT中移除
5. 隐匿RT中所有剩余的元组
6. 将GS中所有分组以QIT，ST形式发布

算法的第1步是预处理阶段，得到后面步骤所需要的元组边；第2步到底10步是分组阶段，整个过程只有一个while循环；得到所有分组后，算法第11步到14步是对RT中的剩余元组的处理，看是否能加入到已有分组中且不影响原有分组的多敏感属性L多样性，从而减小元组的隐匿数量。算法第13/14步是隐匿数据和数据发布阶段。

4.2.3 BES算法实例应用

我们以表4-2-1作为待发布数据表为例，通过BES分组算法进行分组。假设多样性参数L=3。首先得到的类二部图如图4-2-3。

表 4-2-1 待发布数据表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tuple ID | Name | Age | Sex | Zipcode | Physician | Disease |
| t1 | Sam | 23 | M | 821071 | John | Flu |
| t2 | Anne | 44 | F | 821023 | John | Pneumonia |
| t3 | Mike | 56 | F | 821045 | John | Cancer |
| t4 | Lily | 35 | M | 821123 | Bob | Flu |
| t5 | Harry | 25 | F | 821031 | Bob | Pneumonia |
| t6 | Mona | 39 | M | 821035 | Anne | Gastritis |
| t7 | Tony | 40 | F | 821110 | Anne | Gastritis |
| t8 | Lucy | 37 | M | 821115 | Hugo | HIV |
| t9 | Tim | 60 | M | 821134 | Marry | Flu |
| t10 | Lucy | 45 | F | 821002 | John | Flu |
| t11 | Mona | 31 | F | 821134 | Bob | Pneumonia |

图 4-2-3 由表4-2-1映射的类二部图

//TODO

再的到的元组边为TE={t1.E{(John，Flu)}，t2.E{(John，pneumonia)，t3.E{(John，Cancer)}，t4.E{(Bob，Flu)}，t5.E{(Bob，Pneumonia)}，t6.E{(Anne，Gastritis)}，t7.E{( Anne，Gastritis)}，t8.E{(Hugo，HIV)}，t9.E{(Marry，Flu)}，t10.E{(John，Flu)}，t11.E{(Bob，Pneumonia)}}，为能够进行不相交边操作，我们需要得到元组边的边点集V（t.E）,根据定义4.7我们可以得到V（t1.E）={John，Flu}，V（t2.E）={John，Pneumonia}，V（t3.E）={John，Cancer}，V（t4.E）={Bob，Flu}，V（t5.E）={Bob，Pneumonia}，V（t6.E）={Anne，Gastritis}，V（t7.E）={ Anne，Gastritis}，V（t8.E）={Hugo，HIV}，V（t9.E）={Marry，Flu}，V（t10.E）={John，Flu}，V（t11.E）={Bob，Pneumonia }。遍历TE，首先选取ti到当前分组G中，依次遍历找到t5，因为，能够进行不相交边选择操作，所以将t5加入到当前分组G中，继续遍历TE得到t6有且，满足不相交边选择条件，所以讲t6将入到当前分组G，得到一个满足L=3条件的有效分组{t1，t5，t6}，并将它们对应的元组边从TE中移除。重复以上过程继而得到{t2，t4，t7}，{t3，t8，t9}。剩余元组为{t10，t11},处理剩余记录，在不破坏原有分组多敏感属性L-多样性的前提下可将t11加入到分组3中得到{t3，t8，t9，t11}，最后剩余t10隐匿。附加信息损失度为：1/9,只有一条数据被隐匿，所有数据隐匿率为1/11。最终发布数据表如4-2-2所示：

表4-2-2 多敏感L-多样性（L=3）发布结果

QIT（准标识表） ST（敏感属性表）

|  |  |  |
| --- | --- | --- |
| Tuple ID | QIDs | Group ID |
| t1 | ··· | G2 |
| t2 | ··· | G1 |
| t3 | ··· | G3 |
| t4 | ··· | G1 |
| t5 | ··· | G2 |
| t6 | ··· | G2 |
| t7 | ··· | G1 |
| t8 | ··· | G3 |
| t9 | ··· | G3 |
| t11 | ··· | G3 |

|  |  |
| --- | --- |
| Group ID | Sensitive Attribute |
| G1 | <John，Pneumonia> |
| <Bob，Flu> |
| <Anne，Gastritis> |
| G2 | <John，Flu> |
| <Bob，Pneumonia> |
| <Anne，Gastritis> |
| G3 | <John，Cancer> |
| <Hugo，HIV> |
| <Marry，Flu> |
| <Bob，Pneumonia> |

4.3 实验结果及分析

4.3.1 实验数据集

实验实际数据集采用UCI machine learning repository 的人口统计数据集，数据集来自<http://archive.ics.uci.edu/ml/datasets/adult，adult>数据集包含部分美国人人口普查数据。该数据集也是大多是针对关系型敏感数据隐私保护研究的实验实际数据集，已成为数据发布隐私保护研究领域的标准测试数据集。对原始数据集进行处理，过滤掉不完整的记录，进行数据格式转换后提取10K（1K=1000）数据记录，并选取其中五个属性作为敏感属性，如表4-3-1所示。实验硬件环境：Intel Core i5-7200U CPU 2.5GHz，8GB RAM

操作系统平台：Microsoft Windows 10 专业版

实验编程环境：IntelliJ IDEA， Mysql 5.6.24

表4-3-1 实验数据集信息

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 敏感属性 | Occupation | Education | Marital-status | Workclass | Race |
| 基数 | 14 | 16 | 7 | 8 | 5 |

表4-3-2 实验中采用的复合敏感属性

|  |  |
| --- | --- |
| 敏感属性个数 | 复合敏感属性 |
| 2 | <Occupation，Education> |
| 3 | <Occupation，Education，Marital-status> |
| 4 | <Occupation，Education，Marital-status，Work-class> |
| 5 | <Occupation，Education，Marital-status，Work-class，Race> |

4.3.2 实验及结果分析

//TODO

4.4 本章小结

本章主要是针对一般多敏感属性数据发布提出了一种新的基于类二部图的分组算法，定义了算法操作算法思想，并通过举例说明算法的执行过程，最后通过实际数据集来从XXXXXXX//TODOXXXXXXXXXXXXX方面验证了算法的XXX//TODOXX。

1. **面向多敏感属性的个性化发布模型**

在本文第四章中，主要是针对一般性的多敏感数据集进行分组，实现隐私数据的保护，将数据集中所有敏感属性值视为具有相同的隐私度量。在实际应用中，针对不同的场景，不同的敏感属性值，甚至是敏感属性间都可能存在不同的敏感度问题。例如敏感属性“Disease”中敏感值“HIV”的敏感度肯定要高于敏感值“Flu”的敏感度。敏感属性“Disease”属性敏感度要高于敏感属性“Occupation”的属性敏感度。所以，为了适应实际场景下的数据发布，需要考虑针对不同敏感值，不同敏感属性的敏感度问题，制定个性化的数据发布方案。

目前关于个性化敏感属性数据发布隐私保护领域中，主要采用匿名化分组和有损连接发布两种方式。本文继续采用多敏感属性个性化L-多样性模型，现有多敏感属性个性化发布方案中针对敏感属性值的权值进行定义或者作等级划分，都是将记录中的敏感属性值拆分成单个的敏感属性取值，然后再作处理，从而实现单个的高敏感度敏感属性值的个性化保护[2]。或者直接将个体记录作为一个整体，对每个个体记录指定个性化发布约束[9]，这种个性化定制方式虽然对隐私信息的保护较好，但是效率低，可行性不高，且都没有考虑到敏感属性本身的敏感度问题（例如疾病的敏感度大于收入的敏感度）。本章拟在考虑单个敏感属性值的敏感度的同时也考虑个体记录的整体敏感度，由于个体记录的整体敏感度由组成该个体记录的所有敏感属性值的敏感度决定，本章利用结合单个敏感属性的个性化约束与个体整体记录的个性化约束结合的方式，整体个体记录的约束性由个体的每一项隐私属性决定，最后分组时考虑个体记录的敏感度，而数据发布时仅仅需要为单个敏感属性值指定个性化约束，从而达到的隐私保护的目的。

5.1 多敏感属性(L，)-diversity 个性化数据发布模型

为了方便描述与理解，这里依然引述本文第四章中的符号定义。发布关系型数据表T={A1，A2，A3，···，Ap ，S1 S2 S3，···，Sd}，其中Ai{A1，A2，···，Ap}（1ip）为准标识属性，Sj{S1，S2，···，Sd}（1jd）为敏感属性。待发布数据表T中共有n条记录，即|T|=n，数据表中每条个体记录称为一个元组，元组标识为ti（1in）。t[X]表示数据表中元组t在X属性上的取值。

由于需要对待发布数据表中的敏感属性进行个性化隐私保护，这里我们采取对敏感值设定权值（）的方式。并给出以下描述。

5.1.1 相关定义与描述

定义5.1 敏感属性值权值。待发布数据表T中，敏感属性Sj{S1，S2，···，Sd}（1jd）的值域Dom（Sj）中，Dom（Sj）={V1Sj，V2Sj，···，VnSj }，对敏感属性值ViSji（1in具有其对应的敏感度权值，记为(ViSj)，且对，有(ViSj)。

在初始化待发布多敏感属性数据集的时候，需要对数据集中所有敏感属性值根据实际情况个性化地定制其敏感属性权值，并称ViSjVjSj（1ijn）当且仅当在同一敏感属性Sj中，敏感值Vi的敏感度权值低于敏感值Vj的敏感度权值，敏感属性值敏感度权值越大则表明该值敏感度越高。并且注意这里的敏感属性值的敏感度高低比较的前提是指敏感属性值在同一敏感属性下，不同敏感属性的敏感值之间的敏感度比较需要考虑到敏感属性本身的权值。

定义5.2 敏感属性权值。待发布数据表T中，敏感属性SA={S1，S2，···，Sd}，对于敏感属性SjSA，设定个性化的敏感度权值，表示为SWeight(Sj)，且0（Sj）。

关于敏感属性自身的敏感度问题，在目前针对敏感属性的个性化数据发布研究中都是将敏感属性本身看作是具有相同的敏感度，但实际上待发布数据表T中不同的敏感属性应该具有不同的敏感度权值的，例如在表3-1-1中的两个敏感属性中，“Physician”属性的敏感属性权值应该是要低于敏感属性“Disease”敏感属性权值的。

定义5.3 元组边敏感度权值。待发布数据表中T中，元组ti，对应元组边为ti.E。元组边敏感度权值表示为TWeight（ti.E）。元组边敏感度权值综合该元组所有敏感属性取值的权值和其对应敏感属性的敏感度权值，ti元组边权值与其ti上的敏感属性值权值(t[Sj])和敏感属性权值SWeight(Sj)的关系为：

即元组边的敏感度权值为当前元组所有敏感属性值和其对应的敏感属性的敏感度权值乘积之和。在数据发布过程中，只需要根据不同场景设置个敏感属性值的不同敏感度取值和敏感属性的敏感度取值，就可以通过以上关系计算得到所有元组对应的元组边的元组边敏感度权值，利用第四章说明的映射方法即可得到带权的类二部图。以下以待发布数据表4-2-1为例，计算得到所有元组边的元组边敏感度权值。表5-1-1为对待发布数据表中的敏感属性值和敏感属性按照医疗数据隐私保护场景设定的相应铭感度权值。敏感属性“Physician”各敏感值的敏感度由与该医生主治哪些病相关，敏感属性“Disease” 各敏感值的敏感度由该疾病对与个体记录自身的敏感性相关。

表5-1-1 敏感度权值设置

|  |  |  |  |
| --- | --- | --- | --- |
| Physician | VWeight | Disease | VWeight |
| John | 0．7 | Flu | 0．2 |
| Bob | 0．5 | Pneumonia | 0．6 |
| Anne | 0．5 | Gastritis | 0．5 |
| Hugo | 0．9 | HIV | 0．9 |
| Marry | 0．2 | Cancer | 0．9 |

|  |  |
| --- | --- |
| SA | SWeight |
| Physician | 0．3 |
| Disease | 0．7 |

根据定义5.3我们可以得到表4-2-1中个元组对应的元组边敏感度权值。TWeight（t1.E）=0.70.3+0.20.7=0.35, TWeight（t2.E）=0.70.3+0.60.7=0.63,同理依次可求得各元组对应的元组边敏感度权值，最终处理完后得到的元组边集合表5-1-2所示：

表5-1-2 带权元组边集合表

|  |  |  |  |
| --- | --- | --- | --- |
| Tuple ID | SA | t.E | TWeight |
| t1 | < John，Flu > | （John，Flu） | 0.35 |
| t2 | < John，Pneumonia > | （John，Pneumonia） | 0.63 |
| t3 | < John，Cancer > | （John，Cancer） | 0.84 |
| t4 | < Bob，Flu > | （Bob，Flu） | 0.29 |
| t5 | < Bob，Pneumonia > | （Bob，Pneumonia） | 0.57 |
| t6 | < Anne，Gastritis > | （Anne，Gastritis） | 0.52 |
| t7 | < Anne，Gastritis > | （Anne，Gastritis） | 0.52 |
| t8 | < Hugo，HIV > | （Hugo，HIV） | 0.9 |
| t9 | <Marry，Flu > | （Marry，Flu） | 0.2 |
| t10 | <John，Flu > | （John，Flu） | 0,35 |
| t11 | <Bob，Pneumonia > | （Bob，Pneumonia） | 0.57 |

在实际应用中，只要我们确定了所有敏感属性值对应的敏感度权值和敏感属性对应的敏感度权值，通过定义5.3中给定的计算方式即可得到所有元组对应的元组边敏感度权值。该计算方式得到的元组边敏感度不仅仅考虑了敏感值本身的敏感度，而且结合了敏感属性也具有的敏感属性度，综合求得元组边最终的敏感度权值，更好的实现发布数据表的个性化定制。另一方面，由于计算方式给定，只需要给出敏感属性值和敏感属性的敏感度权值就可以自动计算所有元组的元组边敏感度权值，即使存在大量待发布数据时，借助计算机处理也能很简单快速完成处理，所以该方式简单易行，具有较高的实际操作性。

5.1.2 (L，)-diversity 个性化数据发布模型描述

在得到待发布数据表中所有元组对应的元组边敏感度权值后，我们可以在得到的带权元组边集合上进行分组，使得到的分组满足（L，）- diversity个性α化匿名模型。（L，）- diversity个性化匿名模型是建立在多敏感属性L-多样化匿名模型之上的。设最终发布数据集为T’={G1，G2，G3，···，Gm}，Gi（1im）为T’上的一个有效分组，若Gi满足：

（1）Gi中任一敏感属性满足单敏感属性L-多样性原则，且Gi中所有敏感属性满足复合敏感属性L-多样性性质。

（2）Gi中所有元组对应的元组边权值之和小于等于。即有。其中tiGi。若元组边权值之和表示为分组边权值GWeight(Gi),则有

则称分组Gi为满足（L，）- diversity个性化分组。若对T’中所有分组均满足（L，）- diversity个性化分组，则称T’是满足（L，）- diversity个性化隐私数据发布模型，其中代表满足该模型的分组元组边敏感度值之和的最大敏感度阈值。

满足（L，）- diversity个性化数据发布模型的发布数据是隐私安全的，因为（L，）- diversity个性化数据发布模型中的所有分组是建立在复合敏感属性L-多样性的基础之上，所以（L，）- diversity个性化数据发布模型是安全的数据发布模型的证明可参考文献[2]中定理2的证明。本模型是在得到所有元组边敏感度权值的基础上提出，要求安全的分组中必须满足其元组边敏感度小于等于给定的最大敏感度阈值，关于模型中的最大敏感度阈值，给出以下定义：

最大敏感度阈值等于待发布数据表中所有敏感属性对应敏感值的敏感度权值的平均值乘以敏感属性自身敏感度权值，求和之后在乘以一个个性化系数。

敏感属性Sj平均权值表示为：

= (ViSj） (1)

其中Sj表示数据表T中第j个敏感属性，ViSj表示Sj中的第i个敏感值。例如Sj可以是表4-2-1中的敏感属性“Disease”，ViSj可以是“Disease”中的敏感属性值“Flu”。则=(0.2+0.6+0.5+0.9+0.9)/5=0.62。其中Dom(Disease)=5。

最大敏感度阈值：

(2)

其中L和的取值已经确定，代表Sj的敏感属性权值，也可由类似表4-2-2数据计算出来。我们定义为（L，）- diversity个性化数据发布模型中的个性化可变参数。

可变参数的值，在实际应用中可根据数据发布结果、隐私保护效果调节，基准参数为1，其与大小成正相关。的值直接影响模型定义中的值，值越大导致越大，越大就对模型分组满足（L，）- diversity个性化分组的约束越小，可能导致高敏感度的元组大量重复出现在同一分组中，当取值过大，将完全失去这种约束，失去个性化定制发布保护隐私数据的功能。反之，当取值过小，取值越小，对分组中的元组约束过大，可能导致无法得到满足（L，）- diversity个性化的分组较少，需要隐匿过多的数据，失去数据发布的意义。

（L，）- diversity个性化数据发布模型通过元组敏感度最大阈值来限制同一分组中具有高敏感度元组的个数，合理设置个性化参数可避免出现上文所述高敏感度分组出现在同一分组，导致敏感信息倾斜现象，有效避免了同质攻击，且的取值综合考虑敏感属性值和敏感属性的敏感度，且可跟据模型分组效果反馈调节可变参数来得到最佳，从而得到满足要求的个性化隐私数据发布。因此，本章提出的（L，）- diversity个性化数据发布模型具有较好的实用性。

5.2 带权类二部图边选择分组算法-WBES

带权类二部图边选择分组算法（Weight Bigraph-similar Edge Selection）是由本文第四章中提出的BES算法演变得来，是在对元组边进行加权后映射得到带权类二部图上的分组算法。

定义5.4 排斥元组边。在带权类二部图BG中，在为当前分组G选取一条元组边t.E时，满足不相交边选择,但将t.E加入到当前分组后，当前分组的最大敏感度阈值超过，则称元组t为当前分组G的排斥元组边。

由于WBES算法基本思想由BES算法演变而来，下面直接给出WBES的描述。

算法. 带权类二部图的边选择分组算法

输入：待发布数据表T{A1，A2，A3，···，Ap，S1，S2，···，Sd}，多样性参数L

输出：准标识属性表QIT，敏感属性表ST

步骤：

1. 处理T中的敏感属性与敏感属性值，得到敏感属性值集敏感属性的敏感度权值。

2.遍历T中所有元组，得到每个元组的元组边敏感度权值

3．提取T中敏感属性值，构建带权类二部图BG，得到所有记录的带权元组边WTE

1. while WTE中的元组边不为空
2. 遍历WTE-> t[i···n].E
3. if ti.E能在当前分组G上作不相交边选择且ti.E不是排斥元组边
4. 将ti将入到当前分组G
5. if 当前分组G中元组数等于L
6. 将当前分组G加入到已完成分组集合GS中，并将G中所有

元组对应的元组边从WTE中移除

1. 停止遍历，重新进入到while循环
2. if 当前分组G中元组数不足L
3. 将分组G中所有元组对应元组放入待处理元组集合RT，并将其

对应的元组边从WTE中移除

13.遍历RT->t[i···|RT|]

14. for G[j…|GS|]

15. if ti加入到GSj中GSj依然满足多敏感属性（L，）-diversity

个性化分组

16. 元组ti加入到GSj中，并将ti从RT中移除

17.隐匿RT中所有剩余的元组

18.将GS中所有分组以QIT，ST形式发布

我们依然以表4-2-1为待发布数据表，，根据表5-1-1中的敏感度权值设置来执行算法，算法处理第一步得到表5-1-2的带权元组边集合，映射成带权类二部图如5-2-1所示。得到带权元组边集合WTE={//TODO}

图5-2-1 带权类二部图

//TODO

令，则根据表5-1-1敏感度权值的设置和5-1-2中式（2）计算得到，我们为使数据发布达到一定的安全性，取L=3，则我们需要得到分组发布数据表满足（3，）-diversity个性化敏感数据发布表。根据算法第一步，遍历带权元组边集合，选取t1加入当前分组G，且GWeight（G）=0.35，依次遍历带权元组边集合得到t5既能进行不相交元组边选择操作且不是当前分组G的排斥元组边，将t5加入到当前分组得到G={t1，t5}，此时GWeight（G）=0.92。继续遍历带权元组边集合得到t6满足不相交边选择操作并且不是当前分组G的排斥元组边。将t6加入当前分组得到一个满足（3，）-diversity个性化的分组G1={t1，t5，t6}，并且GWeight（G1）=1.44。算法继续循环，最后可得另外两个分组G2={t2，t4，t7}，GWeight（G2）=1.44，G3={t3，t8，t9}，GWeight（G3）=1.94{t10，t11}，剩余元组中虽然t11加入分组G3中能够满足L多样性质，但是t11是t3的排斥元组边，所以不能加入。最后隐匿元组{t10，t11}。最后得到发布表如5-2-1所示：

表5-2-1 （3，）-diversity个性化敏感数据发布表

QIT(准标识属性表) ST(敏感属性表)

|  |  |
| --- | --- |
| Group ID | Sensitive Attribute |
| G1 | <John，Flu> |
| <Bob，Pneumonia> |
| <Anne，Gastritis> |
| G2 | <John，Pneumonia> |
| <Bob，Flu> |
| <Anne，Gastritis> |
| G3 | <John，Cancer> |
| <Hugo，HIV> |
| <Marry，Flu> |

|  |  |  |
| --- | --- | --- |
| Tuple ID | QIDs | Group ID |
| t1 | ··· | G1 |
| t2 | ··· | G2 |
| t3 | ··· | G3 |
| t4 | ··· | G2 |
| t5 | ··· | G1 |
| t6 | ··· | G1 |
| t7 | ··· | G2 |
| t8 | ··· | G3 |
| t9 | ··· | G3 |

5.3 L-拆分带权元组边选择分组算法

在本章5.2节中提出在映射得到带权类二部图的带权元组边集合上利用WBES分组算法得到满足（L，）-diversity个性化因数数据发布表，该算法由于是在BES算法上添加附加条件排斥元组边，得到的数据发布结果，通过表5-2-1的个性化发布数据表我们可以发现在发布表中的G3分组，虽然该分组中的满足（3,1.98）-diversity个性化数据发布模型，但是<John，Cancer>，<Hugo，HIV>两个高敏感度元组边依然被划分到同一分组，在这种发布表中，攻击者只要确定个体记录属于分组G3，就可确定该个体记录大概率（超过1/L）患有HIV或者Cancer。造成隐私泄露，通过分析我们可以看到WBES桶BES一样是根据生成的带权元组边顺序选取元组划分组，由于元组本身是无序的，可能出现高敏感度元组相邻并且划分到同一组，然后选取到较低敏感度的元组以满足GWeight小于等于分组敏感度阈值的条件，从而造成虽然得到的分组虽然满足（L，）-diversity个性化分组，分组内的元组的敏感度权值出现两极分化（例如分组G3中个元组的元组边敏感度权值{0.84,0.9,0.2}），造成隐私泄露。为了避免这种情况，让得到的分组既满足（L，）-diversity个性化分组需求，又使得分组中的元组敏感度权值高低较分布均匀，保护敏感隐私信息，本节提出一种改进的带权元组边选择方法-- L-拆分元组边选择分组算法(L-SWES)

5.3.1 算法基本思想

WBES算法中存在的主要问题是元组的分布问题，由于元组由元组生成，元组边集合中的权值分布随机，造成由于WBES算法顺序选取元组的机制，造成算法分组结果有很大的随机性，为避免这种情况，L-拆分元组边选择分组算法（L-Split Weight Edge Selection）的主要思想是将得到元组边集合按照元组边敏感度权值进行降序排序，然后对元组边集合进行权值L等级划分。

定义5.5 降序带权元组边集合L等级划分。对降序带权元组边集合WTE划分成L个子集合，记作WTE\_Subset={WTE1，WTE2，WTE3，···，WTEL}。WTEi(1iL)称为一个等级子集合。并且有WTE=i。各子集合中的元组边敏感度权值一次递减，表示为：WTE1WTE2，WTE3，···，WTEL

关于带权元组边集合L等级划分如图5-3-1所示。

图 5-3-1 WTE权值等级L-划分

//TODO

WTE\_Subset中前L-1个子集合的大小相同为，最后一个子集合的大小| WTEL |。L-拆分元组边选择分组算法（L-SWES）在进行分组是，每个分组的元组尽量从WTE\_Subset中依次选取，若在一个子集合中没有选取到合适元组则可在后面元组继续选取知道选取到元组或者遍历完所有WTE\_Subset。

5.3.2 算法描述

算法. L-拆分元组边选择分组算法

输入：待发布数据表T{A1，A2，A3，···，Ap，S1，S2，···，Sd}，多样性参数L

输出：准标识属性表QIT，敏感属性表ST

步骤：

1. 处理T中的敏感属性与敏感属性值，得到敏感属性值集敏感属性的敏感度权值
2. 遍历T中所有元组，得到每个元组的元组边敏感度权值
3. 提取T中所有带权敏感属性，然后构建带权类二部图，得到带权元组边集合WTE
4. 对WTE进行降序排序
5. while WTE中的元组边不为空
6. 划分WTE得到子集合WTE[1…L]
7. for i=1->L
8. if (tj = WTE[i]中某元组边对应元组)能加入当前分组G
9. tj加入当前分组G
10. if 分组G中元组数等于L
11. 将当前分组G加入到已完成分组集合GS中，并将G中

所有元组对应的元组边从WTE中移除，清空G

1. 重新进入while循环
2. if 当前分组中元组数不足L
3. 将分组G中所有元组放入待处理元组集合RT，并将其对应的元

组从WTE中删除，清空G

1. if WTE中的元组数量小于L
2. 将WTE中所有剩余元组全部加入到RT中，清空WTE中元组
3. 遍历RT
4. for G[j···|GS|]
5. if ti加入到当前分组Gj中依然满足（L，）-diversity个性

化分组

1. 元组ti加入到GSj中，并将ti从剩余元组集合RT中移除
2. 隐匿RT中所有剩余的元组
3. 将GS中所有分组以QIT，ST形式发布

5.3.3 L-SWES算法实例

这里由于L-SWES算法是对WBES算法的改进，为了显示两种算法的对比性，我们依然以表4-2-1为待发布数据表，依然取1.1，则（L，）-diversity个性化数据发布模型中，L=3，我们需要根据L-拆分带权元组边选择算法得到满足（3,1.98）-diversity个性化敏感数据发布表。根据算法1、第2步得到排序后的元组边集合，并将其进行L划分为逻辑上L个元组边子集。得到表5-3-1。

表5-3-1 降序元组边集合（WTE）表

|  |  |  |  |
| --- | --- | --- | --- |
| Tuple ID | SA | TWeight | WTEi |
| t8 | <Hugo，HIV> | 0.9 | WTE1 |
| t3 | <John，Cancer> | 0.84 |
| t2 | <John，Pneumonia> | 0.63 |
| t5 | <Bob，Pneumonia> | 0.57 | WTE2 |
| t11 | <Bob，Pneumonia> | 0.57 |
| t6 | <Anne，Gastritis> | 0.52 |
| t7 | <Anne，Gastritis> | 0.52 |
| t1 | <John，Flu> | 0.35 | WTE3 |
| t10 | <John，Flu> | 0.35 |
| t4 | <Bob，Flu> | 0.29 |
| t9 | <Marry，Flu> | 0.2 |

根据算法，在得到带权元组边集合后，首先选取WTE1中t8加入到当前分组G中，然后跳到WTE2中选取t5能够加入到当前分组G（能做不相交边选择且t5元组对应元组边不是当前分组的排斥元组边）,然后跳到WTE3中选取t1，得到当前一个有效分组G1={t8，t5，t1},且GWeight（G1）=1.82,算法继续循环，同理可得G2={t3，t11，t7}，且GWeight(G2)=1.93,G3={t2，t6，t4}，且GWeight（G3）=1.44。剩余元组{t10，t9}，t9虽然能加入到G2中满足多敏感属性L-多样性，但是不满足分组敏感度权值GWeight，不能满足（L，）-diversity个性化分组要求，所以元组t10，t9需要隐匿。最终得到如表5-3-2的最终数据发布表。

表5-3-2 L-WSES分组算法得到最终数据发布表

|  |  |
| --- | --- |
| Group ID | Sensitive Attribute |
| G1 | <Hugo，HIV> |
| <Bob，Pneumonia> |
| <John，Flu> |
| G2 | <John，Cancer> |
| <Anne，Gastritis> |
| <Bob，Pneumonia> |
| G3 | <John， Pneumonia > |
| < Bob，Flu > |
| <Anne，Gastritis> |

QIT(准标识属性表) ST(敏感属性表)

|  |  |  |
| --- | --- | --- |
| Tuple ID | QIDs | Group ID |
| t1 | ··· | G1 |
| t2 | ··· | G3 |
| t3 | ··· | G2 |
| t4 | ··· | G3 |
| t5 | ··· | G1 |
| t6 | ··· | G3 |
| t7 | ··· | G2 |
| t8 | ··· | G1 |
| t11 | ··· | G2 |

根据最终发布表我们可以得到各有效分组中元组边敏感度权值分布，G1{0.9,0.57,0.35}，G2{0.84,0.57,0.52}，G3{0.63,0.52,0.29}，可以看出分组中的元组敏感度分布高低较为均匀，且没有出现高敏感度元组出现在同一分组的情况，因此L-SWES算法在对WBES算法进行改进后，采取按元组敏感度分段取元组的方式，有效解决了高敏感度元组出现在同一组，敏感信息倾斜现象，保护了数据隐私。

5.4 实验及结果分析

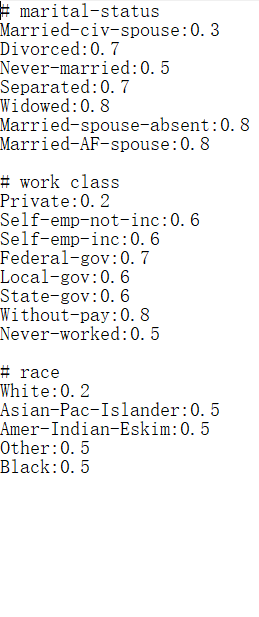
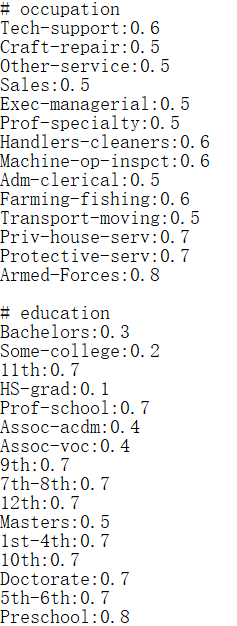
5.4.1 实验数据

实验数据集依然采用本文第四章中采用的美国人口普查数据Adult数据集，硬件环境相同，数据处理与数据结构与第四章相同。由于本章算法讨论的是基于带权值的个性化数据发布，在第四章已有数据集上我们需要定义个敏感属性和敏感属性值的具体权值大小，各权值取值具体定义见表5-4-1与5-4-2。

表5-4-1 不同敏感属性个数下敏感属性自身敏感度设定值

|  |  |  |
| --- | --- | --- |
| 敏感属性个数 | 敏感属性 | 对应权值 |
| 2 | <Occupation，Education> | [0.5，0.5] |
| 3 | <Occupation，Education，Marital-status> | [0.3，0.3，0.4] |
| 4 | <Occupation，Education，Marital-status，Work-class> | [0.2，0.2，0.4，0.2] |
| 5 | <Occupation，Education，Marital-status，Work-class，Race> | [0.2，0.2，0.3，0.2，0.1] |

表5-4-2 各敏感属性值敏感度设定值

5.4.2 实验结果分析

beita 数据隐匿率 时间运行效率 参考其他

//TODO

5.5 本章小结

本章主要展开的是对多敏感属性个性发布模型及其算法实现的讨论。在总结现有多敏感属性数据发布中存在的敏感信息倾斜现象等提出了一种新的多敏感属性个性化隐私数据发布模型，并在第四章提出的算法的基础上针对带权敏感属性提出带权的分组算法，并对带权分组算法提出改进，得到比较好的多敏感数据个性化隐私数据发布效果。

1. **总结与展望**

针对什么问题提出了什么样的模型与算法，达到了什么样的效果

非结构化数据集的数据发布的隐私保护等

**参考文献**

1. X. Xiao and Y. Tao. Anatomy: Simple and effective privacy preservation. in: the 32nd International Conference on Very Large Data Bases (VLDB’06). Seoul, Korea. 2006. 139~150
2. 杨晓春, 王雅哲, 王斌, 等. 数据发布中面向多敏感属性的隐私保护方法. 计算机学报,2008,31(4):574-587.
3. Machanavajjhala A,Kifer D,Gehrke J. et al. l-diversity: privacy beyond k-anonymity[C]// Proceedings of the 22nd International Conference on Data Engineering(ICDE’06). New York: ACM,2006:24-35
4. 孙岚, 郭旭东, 王一蕾, 吴英杰. 个性化隐私保护轨迹发布算法[J]. 系统工程与电子技术.2014,36(12):2550-2555
5. Machanavajjhala A，Gehrke J，and Kefer D．／-diversity：Privacy beyond k-anonymity//Proceedings of the 22nd International Conference on Data Engineering．Atlanta，Georgia,
6. 面向多敏感属性医疗数据发布的隐私保护技术 – 金华 刘善成等人
7. Xiao Xiaokui,Tao Yufei. Personalized Privacy Preservation[C] / / Proceedings of ACM SIGMOD International Conferenceon on Management of Data.New York,USA,2006:229-240.
8. 韩建民, 于娟, 虞慧群, 贾泂. 面向敏感值的个性化隐私保护[J]，电子学报,2010,38(7):1723-1728P.
9. 一种基于最小选择度优先的多敏感属性个性化l-多样性算法。杨静等人
10. 龚奇源，杨明，罗军舟．面向关系一事务数据的数据匿名方法
11. Qiyuan Gong，Ming Yang，Zhenguo Chen，Junzhou Luo．Utility Enhanced Anonymization for Incomplete Microdata
12. Sweeney L．K—anonymity：A model for protecting privacy．International Journal on Uncertainty，Fuzziness，and Knowledge-Based Systems，2002，10(5)：557—570
13. Samarati P，Sweeney L．Generalizing data to provide anonymity when disclosing information