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

第一章 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。其中包括泛化[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[26]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[27]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[30]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（α，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[31]等人提出一种完全（α，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

本文共六章，基本内容如下：

第一章，主要介绍隐私保护的研究背景及意义。说明隐私保护的重要性及隐私保护的研究现状。

第二章，主要是对隐私保护中的相关技术进行了介绍，介绍了隐私保护研究中主要采取的隐私保护方法，最常用的数据发布模型和隐私保护中常用技术即研究的发展方向，分析了这些模型解决了哪些隐私保护中存在的问题和个模型依然存在的缺陷。

第三章，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。

第四章，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验证了算法的有效性。

第五章，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提出对带权敏感属性数据的分组算法，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。

第六章，主要是对本文研究工作的总结，和对未来在个性化隐私保护数据发布研究方向的展望。

第二章 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个人隐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

Tuple ID	Name	Age	Zipcode	Disease
t1	Sam	23	821071	Flu
t2	Anne	44	821023	Pneumonia
t3	Mike	56	821045	Cancer
t4	Lily	35	821123	Flu
t5	Harry	25	821031	Pneumonia
t6	Mona	30	821035	Gastritis
t7	Tony	40	821110	Gastritis
t8	Lucy	37	821115	HIV

（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{23，35，44，56}，可泛化为{[21,30]，[31,40]，[41,50]，[51,-]};邮政编码属性值{821071，821023，821045，821123，821135，821115}，泛化为{8210**，8210**，8210**，8211**，8211**，8211**)。得到泛化后的数据表如表 2-1-1。

表 2-1-1 准标识属性泛化表

Tuple ID	Name	Age	Zipcode	Disease
t1	Sam	[21,30]	8210**	Flu
t2	Anne	[41,50]	8210**	Pneumonia
t3	Mike	[50,-]	8210**	Cancer
t4	Lily	[31,40]	8211**	Flu
t5	Harry	[21,30]	8210**	Pneumonia
t6	Mona	[21,30]	8210**	Gastritis
t7	Tony	[31,40]	8211**	Gastritis
t8	Lucy	[31,40]	8211**	HIV

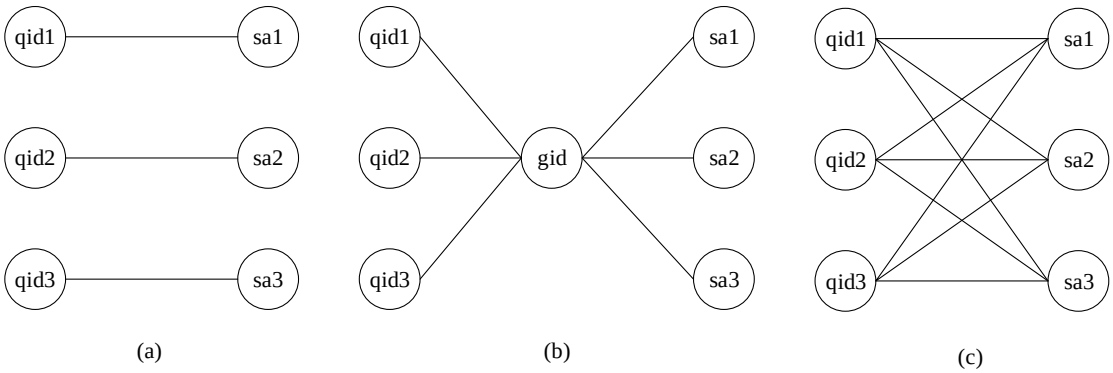
经过变形的数据相对于原始数据含有较少的信息，这样既能够较好地保持数据原有的统计特性，又能保证数据的实用性，当然实际发布中会为隐匿类似“Name”这样的标识属性。泛化是完全不显示部分（或所有）记录的一些属性值。这样会使匿名表中的信息含量降低，但是在某些情况下能够减少泛化数据的损失，达到相对较好的匿名效果。对于泛化方法，Fung 等人[20]总结了五种泛化模式：全域泛化模式、子树泛化模式、兄弟泛化模式、单元泛化模式和多维的泛化。这五种泛化模式中，第一种是搜索空间最小的泛化模式，这种模式下的匿名化数据的变形度(信息损失)是最大的，第四种是搜索空间最大的泛化模式，这种模式下的匿名化数据的变形度(信息损失)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 10 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[26]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



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

普通的去除标识符的匿名方法在链接到外部知识时会造成敏感属性泄露，即发生链接攻击。链接攻击是由于数据发布方不能确定数据接收方存在什么样的背景知识，虽然对发布数据经过一定的处理，但是恶意的数据接收者通过一种将已发布的数据与外部知识链接，从中获取隐私信息的常用攻击手段[23]。其核心思想是：攻击者通过将已发布的数据与已经获知的相关数据进行链接，推理出隐私信息，从而引起信息泄露。数据表的 k-匿名化[24-25]是数据发布时保护私有信息的一种重要方法，是 1998 年由 P.Samarati 和 L.Sweeney 提出的。

k-匿名模型的思想是：在数据发布前对数据进行处理，使得发布的数据中每个元组都存在一定数量(至少为 k 个)的、在准标识属性上取值相同的元组。这样，即使攻击者通过与其他数据进行连接也无法唯一的标识出各元组所有者的身份，仅能以不超过 1/k 的概率标识元组所属个体的身份，从而降低了隐私泄漏的风险。k-匿名（k-anonymity）的具体定义为，给定数据数据表 T，当且仅当在数据表 T 中准标识符属性上每个值序列在等价类中出现次数大于等于 k 时，数据表 T 满足 k-匿名模型。如下表是将表 2-1 匿名化处理后发布的数据，其满足 k-匿名模型（k=3）

表 2-2-1 满足 3-匿名模型的发布表

Tuple ID	Name	Age	Zipcode	Disease
t1	Sam	[21,30]	8210**	Flu
t5	Harry	[21,30]	8210**	Pneumonia
t6	Mona	[21,30]	8210**	Gastritis
t4	Lily	[31,40]	8211**	Flu
t7	Tony	[31,40]	8211**	Gastritis
t8	Lucy	[31,40]	8211**	HIV

该模型中的 k 值是由用户定义的整型参数，通过调整参数 k 的大小可达到对隐私不同程度的保护。k 的大小与隐私保护程度具有如下的关系：k 值取得越大，隐私保护的程度越强。对数据进行处理使其满足 k-匿名，可定会产生一定量的信息损失。因此最终发布数据相对于原始数据信息的损信息失可作匿名化数据质量的一种度量。k 的大小与 k-匿名化数据质量的关系是：k 值取得越大，发布数据中的信息损失量越大，匿名化数据质量也就越低。所以，在实际使用中

时应权衡隐私保护和数据质量两方面的需求，合理选取 k 的值。k-匿名模型提出后，国内外学者在此基础上进行了大量的研究，文献[14]中提出，k-匿名模型对于隐私保护存在许多的安全缺陷：

（1）根据 k-匿名模型得到的的准标识符等价类在敏感属性的取值上上缺乏多样性，若同一等价类分组中的敏感信息相同或者大部分敏感属性取值相似，则容易发生同质攻击。

（2）k-匿名模型得到的发布数据表没有对敏感属性值的敏感度进行区别保护，容易出现高敏感度信息大量出现在同一等价类分组，造成敏感信息的倾斜。在实际的数据处理过程中，敏感属性值的敏感度应该是个性化的，不同的敏感信息值所对应的敏感度强弱应该不是完全相同的。例如在表 2-1 中的 Disease 属性取值中，HIV 与 Cancer 的敏感度肯定是要比 Flu 要高的，因为人们并不介意其他人知道自己患了感冒，然而一般更在意别人知道自己患有癌症。因此，对数据表的处理中，我们更应该加强对高敏感度隐私信息的保护。

2.2.2 L-diversity 模型

由于 k-匿名模型中通过对准标识属性进行泛化处理然后得到等价类分组，仅仅考虑了准标识属性的约束，而缺乏对敏感属性的约束，造成 k-匿名化后的数据仍然可能遭受同质攻击和背景知识攻击，为了解决 k-匿名模型不能抵制同质性攻击和背景知识攻击的问题，Machanavajjhala 等人在 k-匿名模型的基础上提出 L-diversity 匿名模型[14]。L-多样性是基于降低数据表示粒度以达到匿名保护的隐私保护方法。这种降低是一个折中，虽然会导致数据管理或挖掘算法的有效性的一些损失，但也提高了隐私保护。L-多样性模型是 k-匿名模型的扩展，k-匿名使用泛化和抑制降低数据表示的粒度，使得任何给定的记录映射到其所在数据集上至少有 k-1 条其他记录。L-多样性模型能够处理一些 k-匿名模型存在的弱点，特别是当在一些等价类分组敏感属性缺乏多样性时。

L-多样化(L-diversity)定义：数据表 T 中，如果任意等价类中任意敏感属性取值的出现概率不超过 1/L，则称 T 满足 L-多样性模型。

在 k-匿名准标识符等价类分组的基础上，L-diversity 匿名模型还要求等价类分组的敏感属性值至少存在 L 个不同的值，使得攻击者推断出某条记录隐私信息的概率将低于 1/L，从而可以避免攻击者通过同质性攻击来识别敏感信息。例如，表 2-3-1 发布的数据中每个匿名分组都含有三个不同的敏感属性值，因此也是满足 3-多样性的。

表 2-3-1 满足 3-diversity 的发布数据表

Tuple ID	Name	Age	Zipcode	Disease	Group ID
t1	Sam	[21,30]	8210**	Flu	G1
t5	Harry	[21,30]	8210**	Pneumonia	
t6	Mona	[21,30]	8210**	Gastritis	
t4	Lily	[31,40]	8211**	Flu	G2
t7	Tony	[31,40]	8211**	Gastritis	
t8	Lucy	[31,40]	8211**	HIV	

L-diversity 模型解决了 k-匿名存在的可能会受到同质性攻的问题，由于仅考虑了同一分组中敏感属性的分布问题，依然存在以下两个比较明显的缺陷：

（1）L-diversity 模型无法避免可能会出现的高敏感度隐私信息分布的倾斜，高敏感度隐私信息的取值可能不同，如 Disease 中的 HIV 和 Cancer，按照 L-diversity 模型描述，无法避免他们出现在同一等价类分组中，当高敏感度隐私属性值大量出现在同一等价类分组时，会造成隐私信息泄露。针对 l-diversity 模型存在的这一问题，在基于 L-diversity 的基础上个性化数据发布模型的研究被大量提出。

（2）L-diversity 模型不能防止概率推理攻击，由于一个等价类分组中的某些敏感属性很自然的比其他敏感属性的频率高，这使得攻击者能够得知组中的某一记录很有可能拥有该属性值。

（3）L-diversity 模型要求较高，当敏感属性分布不均匀时，实现起来困难，甚至难以实现。这也是 l-多样性的局限性所在，它假设各种敏感属性值的频度是相似的。

2.2.3 t-closeness 模型

由于 k-匿名和 L-多样性两种匿名模型在保护数据隐私上存在各自的不足，Li Ninghui[16]等人提出了一个新的隐私保护匿名模型 t-closeness 匿名模型，t-邻近模型的主要思想和 L-diversity 模型的主要思想相似，都是针对同一等价类分组中的敏感属性的分布作处理。t-邻近模型要求敏感属性中的敏感值在等价组中的分布与其在整个原始表中的分布接近。t-邻近模型可以看做是对 L-多样性匿名模型的进一步改进，t-邻近模型通过对数据表中各敏感属性数据值在整个数据集中的整体分布进行分析，而后要求敏感属性在同一等价类分组中的敏感属

性也近似满足该敏感属性在整个数据集中的整体分布。即发布的数据集要在满足k-匿名化模型的基础上，还要求敏感属性值在等价
类内的分布与其在隐私化表中的全局分布的差异低于设定阈值 t，那么这个分组就满足 t-closeness 匿名化，如果所有的分组都满足 t-closeness 匿名化，那么该发布的整个数据表 T 就满足 t-邻近模型。

t-closeness 定义：数据表 T 中，如果任意等价类中敏感属性的数据
分布与数据集 T 的分布差异不超过阈值 t，则称数据集 T 满足 t-closeness 模型。

t-closeness 在 L-多样性模型的基础上，对敏感属性的分布问题提出了要求，它要求所有等价类中敏感属性值的分布与该属性的的全局分布尽量相近。敏感属性在等价类中的分布与在整个表的分布之间的差异，一般采用的是 EMD 距离来衡量两种敏感属性分布的接近程度，并要求这种差异不超过 t。

t-closeness 最主要的优点是很大程度上防止了针对敏感属性值的偏斜性攻击和相似性攻击现象的发生。Li Ninghui 在文献中指出 t-closeness 对属性泄露的保护，但不涉及身份披露。因此，可能需要在同一时间使用 t-closeness 和 k-anonymity 两种隐私保护策略。t-closeness 也存在不足和缺陷：

（1）t-colseness 涉及均匀性，无法保证对 k-anonymity 的背景知识攻击永远不会发生；

（2）t-colseness 隐私化的结果是导致数据发布后的数据的可用性降低，因为它要求相同等价类中的敏感属性分布相同或相近，增大阈值 t 是可以使发布数据可用性提高的唯一有效的方法。

2.3 本章小结

本章主要是对隐私保护中的相关技术进行了介绍，主要是介绍隐私保护研究中主要采取的隐私保护方法，介绍了最常用的泛化和有损连接两种数据发布方式。然后介绍并分析了隐私保护中常用技术即研究的发展方向，包括最早提出的 k-匿名模型，l-多样性模型和再此基础上改进的 t-邻近模型，并分析了这些模型解决了哪些隐私保护中存在的问题和个模型依然存在的缺陷。

第三章 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	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[14]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[35]，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[27] 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{A_1，A_2，A_3，•••，A_p，S_1，S_2，•••，S_d}，其中 A_i（1≤i≤p）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（1≤j≤d）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（1≤i≤n），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（S_i）为 S_i 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[27]. 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. 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 $\bigcup_{j=1}^{m} Gj$=T，并且Qi∩Qj=∅（1≤i≠j≤m）。

（4）单敏感属性 L-多样性[14]。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$\frac{c(v)}{|G|} \leq \frac{1}{L}$，|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（1≤i≤d）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（1≤i≤m）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[29]。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 $t[S_i]$（$t \in G$，$1 \leq i \leq d$），需要将 G 中所有包换 $t[S_i]$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 v 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，$t.TSID=\sum_{i=1}^{d} SID(t[Si])$。对于一分组 G_i，G_i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 $G_i.GSID$，$G_i.GSID=\sum_{i=1}^{|Gi|} ti.TID$。

（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}，$|G_i| \geq L$，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v_i（$1\leq i\leq n$），v_i在G的出现的频率记为$C(v_i)$，则分组G_i的概率泄露度定义为$leak(Gi)=\sum_{i=1}^{n}\left(\frac{c(v_i)}{|G|}-\frac{I}{L}\right)$,则数据集的平均概率泄露度为$leak(T)=\frac{\sum_{i=1}^{m} leak(G_i)}{|T|}$，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 中的每一条记录看作是一个分组,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$\frac{|(G,v)|}{|G|} \leq a$，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 a_v（$0 \leq av \leq 1$），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（a_v，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 a_v 就应该越小；敏感属性值的敏感度越弱，则对应的 a_v 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 a_{HIV} 应该小于“Flu”的频率约束值 a_{Flu}。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 $a_{HIV}=0.4$，$a_{Flu}=0.6$，$a_{Pneumonia}=0.4$，$a_{Gastritis}=0.4$。

表 3-3-1 完全（a_v，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 a_s 的设定原则是，设定值 a_s 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，v_s为敏感属性的取值， a_{vs}为敏感属性 v_s 的频率约束，则 a_{vs}应该满足如下关系式：

$$a_{vs} \geq \frac{|\{t|t[s] = v_s\}|}{|T|}$$

（2） 基于最小选择度优先的多敏感属性分组算法

杨静等人[32]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）={LS_1，LS_2，LS_3，•••，LS_m}。若 CG（S）满足以下关系：$\bigcup_{i=1}^{m} LS_i$=Dom（S）且 $S_i \cap S_j = \emptyset$（$1 \leq i \neq j \leq m$），则就称 CG（S）为 S 的一个值域等级划分。且 SDegree（LS_i）表示敏感属性 S 的在等级 LS_i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：

$$TDegree（t）=\sum_{i=1}^{d} SDegree（t[S_i]）$$

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（t_j）为 t_j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（t_j）表示为：

$$Select（t_j）= \sum_{v \in SSet（tj）} f(v) \times TDeGree(t_j)$$

其中$f(v)$为敏感属性v在待发布数据表中出现的频率。SSet（t_j）是待发布数据表

中个体记录 t_j 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。

第四章 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[14]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[14]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[31]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	qid_1	sa_1
t2	qid_2	sa_2
t3	qid_3	sa_3
•••	••	•••

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{qid_1，qid_2，qid_3}和敏感属性{sa_1，sa_2，sa_3}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	qid_1	G_1
t2	qid_2	
t3	qid_3	
•••	••	•••

Group ID	Sensitive Attribute
G_1	sa_1
	sa_2
	sa_3
•••	•••

4.2 基于类二部图边选择的多敏感数据分组算法--BES

设待发布关系型数据表 T={A_1,A_2,A_3，•••,A_p ,S_1 S_2 S_3，•••,S_d}，其中 A_i∈{A_1，A_2，•••，A_p}（1≤i≤p）为准标识属性，S_j∈{S_1，S_2，•••，S_d}（1≤j≤d）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 t_i（1≤i≤n）。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[14]。将待发布数据表 T 中所有元组分为若干组，记为 GT，GT={G_1，G_2，•••，G_m}，并且 $\bigcup_{j=1}^{m} Gj$=T，并且Qi ∩ Qj=∅（1≤i≠j≤m）。则称 GT 为待发布数据 T 的分组。

定义 4.2 复合敏感属性[27]。待发部数据表中的所有敏感属性组成复合敏感属性 S={S_1，S_2 •••，S_d}，S 为复合敏感属性集合，则|S|≥2。S_i∈S（1≤i≤d）标识数据表中的第 i 个敏感属性，Dom（S_i）表示敏感属性 S_i 的值域，|S_i|标识 Dom（S_i）的基数，即 S_i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。v_i（$1 \le i \le |SSet(G)|$）为分组 G 中某一敏感值，v_i∈SSet(G),c(Vi) 标识 v_i 出现的次数，若分组 G 中满足：$\frac{\max(c(v_i))}{|G|} \le \frac{1}{L}$，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 算法基本思想

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。BES 分组方法首先需要将多敏感属性数据表 T 中所有元组映射到类似二部图结构的图上。

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）。

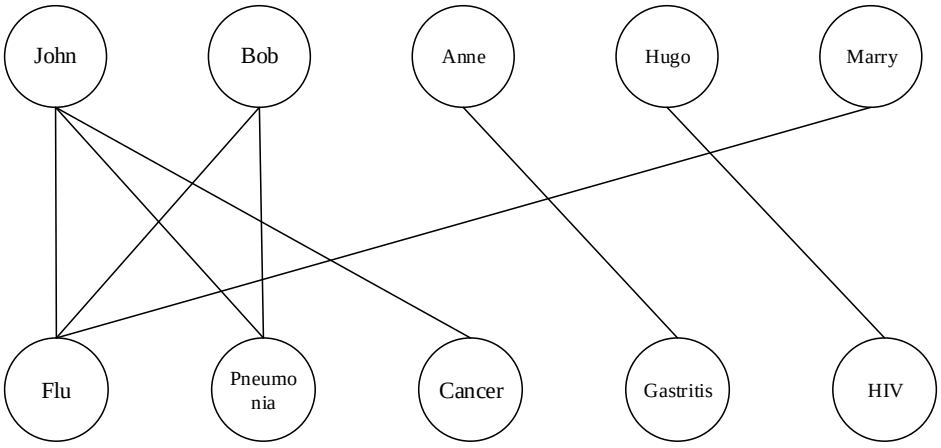
针对待发布医疗数据表 3-1-1 中,共有两个敏感属性 Physician 与 Disease，取值集合分别为：

$S_{Physician}$={John，Bob，Anne，Hugo，Marry}

$S_{Disease}$={Flu，Pneumonia，Cancer，Gastritis，HIV}

其中 $S_{Physician}$∩$S_{Disease}$ =∅,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：

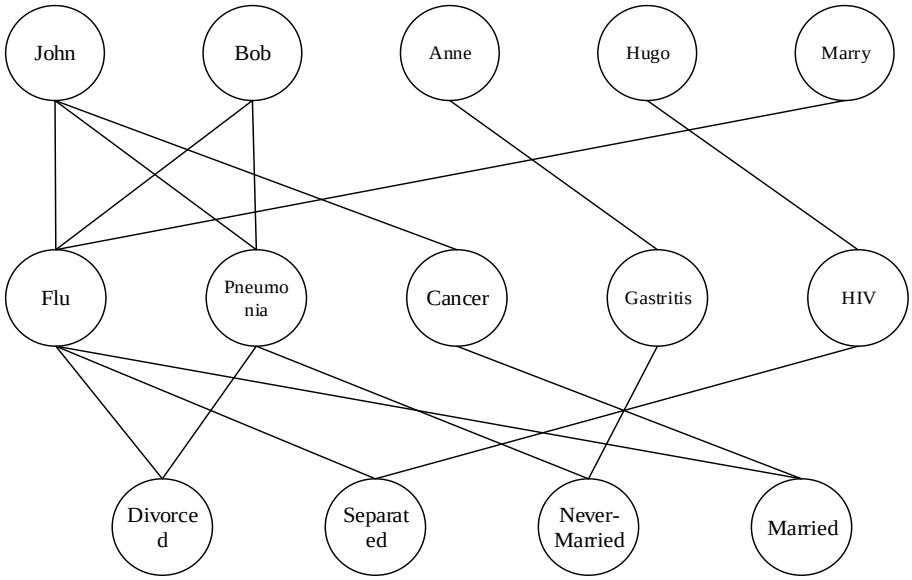
图 4-2-1 表 3-1-1 数据表敏感属映射的类二部图



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 $S_{Marital-status}$={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 $t_i(1\leq i\leq n)$，$t_i\in T$。由 t_i 映射到类二部图的所有边称为该元组的元组边，记为 $t_i.E$。$V(t_i.E)$表示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 $TE=\bigcup_{i=1}^{n} t_i.E$。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（t_1.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=<V_S,TE>，其中 V_S={V_{S1}，V_{S2}，•••，V_{Sd}}，其中 V_{Si}（$1\leq i\leq d$）表示敏感属性 S_i 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 $V_{Disease}$={ 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 V_{ti} 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只出现一次。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[27]定义的附加信息损失度=$\sum_{i=1}^{m}(|Gi|-L)/mL$，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=<V_S,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 $V(t.E) \cap \forall V(tg.E) = \emptyset(tg \in G)$，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 $SuppRatio = Ns/|T|$，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{A_1，A_2，A_3，•••，A_p，S_1，S_2，•••，S_d}，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 t_i.E 能在当前分组 G 上作不相交边选择
5. 将 t_i 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

8. 停止遍历，重新进入到 while 循环

9. if 当前分组 G 中元组数不足 L

10. 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

11. 遍历 RT->t[i ••• |RT|]

12. for G[j…|GS|]

13. if t_i 加入到 GS_j 中 GS_j 依然满足多敏感属性 L-多样性

14. 元组 t_i 加入到 GS_j 中，并将 t_i 从 RT 中移除

15. 隐匿 RT 中所有剩余的元组

16. 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 步到第 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。

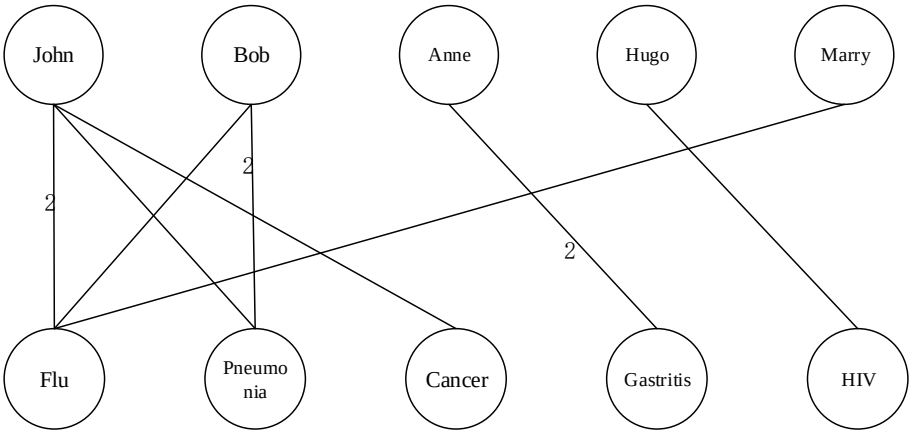
4.2.3 BES 算法实例应用

我们以表 4-2-3 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 4-2-3 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 4-2-3 映射的类二部图



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$V(t1.E) \cap V(t5.E) = \emptyset$，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$V(t1.E) \cap V(t6.E) = \emptyset$且$V(t5.E) \cap V(t6.E) = \emptyset$，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：$(4-3)/3\times 3=1/9$,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 4-2-4 所示：

表 4-2-4 多敏感 L-多样性（L=3）发布结果

QIT（准标识表）

Tuple ID	QIDs	Group ID
t1	•••	G_2
t2	•••	G_1
t3	•••	G_3
t4	•••	G_1
t5	•••	G_2
t6	•••	G_2
t7	•••	G_1
t8	•••	G_3
t9	•••	G_3
t11	•••	G_3

ST（敏感属性表）

Group ID	Sensitive Attribute
G_1	<John, Pneumonia>
	<Bob, Flu>
	<Anne, Gastritis>
G_2	<John, Flu>
	<Bob, Pneumonia>
	<Anne, Gastritis>
G_3	<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

分组算法得到的最终发布数据的隐私安全性由多敏感属性 L-多样的性质保证，因此我们通过 BES 算法的到的发布数据一定是安全的，下面实验主要是通过 BES 算法得到的发布数据进行数据隐匿率和附加信息损失度来评估算法的性能，并从这两个方面在实验数据相同的情况下比较 BES 与文献[27]中提出的基于多维桶的分组算法(MBF)进行对比，综合分析多敏感属性下 BES 分组算法的发布数据的信息缺失和算法性能的优劣。

首先测试不同数据量对数据隐匿率的影响。图 4-3-1 给出了 L-多样性参数 L=3，敏感属性维数 d=3 的情况下待发布数据量从 1K-10K（1K=1000）时 BES 算法与 MBF 算法数据隐匿率的对比。可以看到基于多维桶的分组算法 MBF 和基于元组边选择的 BES 分组算法的数据隐匿率都不高，且随着数据量的增大，两种算法的数据隐匿率都呈下降趋势，这是因为随着数据量的增大，各敏感属性的取值个数越倾向于分布均匀，进而能够得到的有效分组越多，导致实验结果的隐匿率下降。图 4-3-2 给出了在相同条件下，算法的附加信息损失度随着数据量的变化，可以看到随着数据量的增大，附加信息损失度呈下降趋势，这也是因为由于数据量的增加，算法能得到较好的分组，从而导致附加信息损失度的降低。结合图 4-3-1 与图 4-3-2 可以看出，本文提出的 BES 算法数据隐匿率和附加信息损失度保持在一个较低的水平，在进行分组时具有较好的性能，能够保证发布质量较高的发布数据。

图 4-3-1 算法数据隐匿率随着数据量的变化

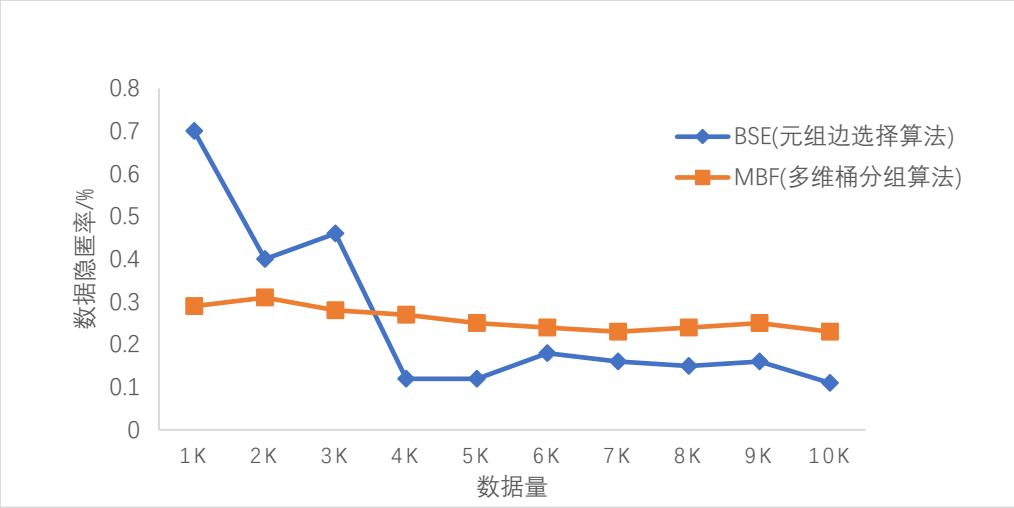
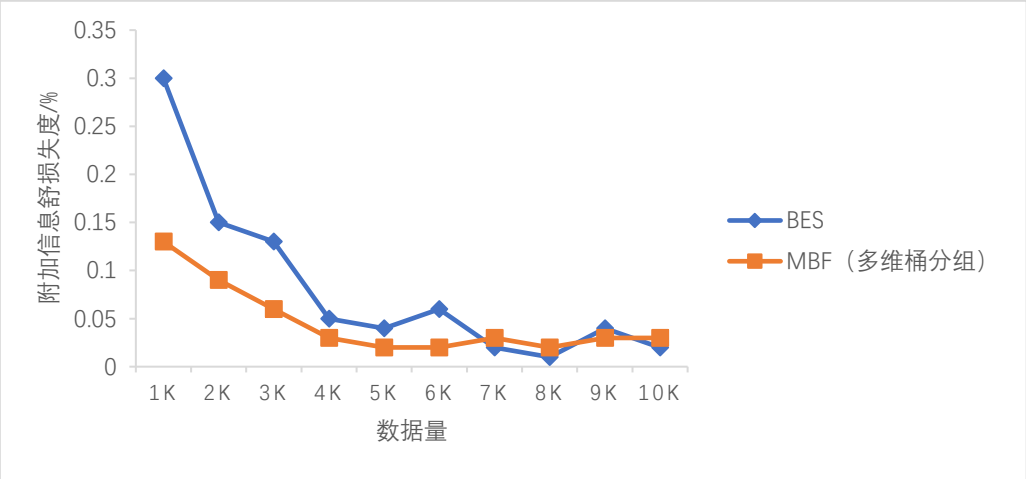
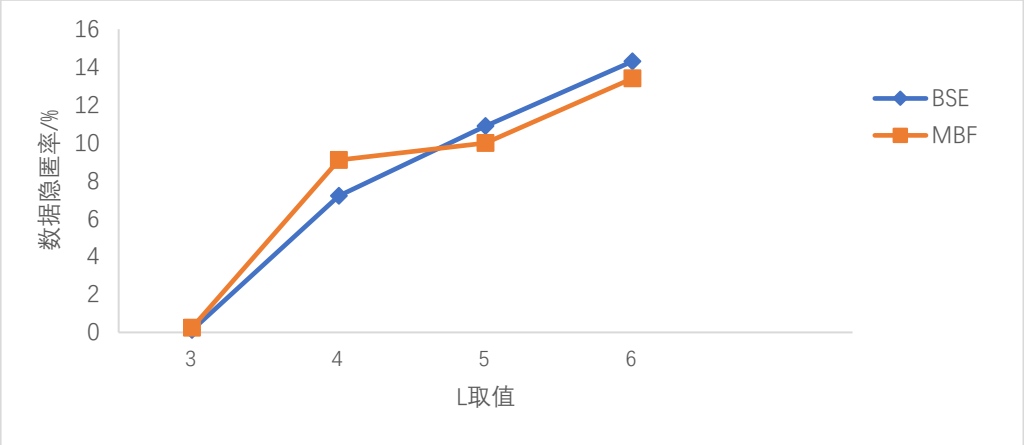


图 4-3-2 算法数据附加信息损失度随着数据量的变化



实验进而对 L-多样性中多样性参数 L 取值变化对数据发布数据隐匿率的影响，图 4-3-3 是 BES 分组算法与 MBF 分组算法在数据量 n=5K，复合敏感属性维数 d=3 时进行的实验测试结果，可以看到随着多样性参数 L 值的增大，两种算法的数据隐匿率都呈明显的上升趋势，当 L 的值小于 4 时，算法的数据隐匿率低于 1%，具有较好的性能。当 L 的值大于 6 时，两种算法的数据隐匿率迅速增加，这是因为数据集中敏感属性 Marital-status 的敏感值取值个数为 7，当多样性参数越接近这个值，在 Marital-status 这一维属性上要保证分组满足 L-多样性越困难，从而造成算法的整体分组效果降低，导致数据隐匿率上升。

图 4-3-3 算法数据隐匿率随着多样性参数 L 的变化



通过大量实验 BES 算法执行时间，在数据量 n=5k，多样性参数 L=3 的情况下，得到在不同敏感属性个数时 BES 分组算法的平均运行时间如表 4-3-3 所示，执行时间只计算算法分组时间和剩余记录处理的时间。可以看到随着敏感属性维度的增加，BES 分组算法的执行效率并没有随着敏感属性维数的增加而降低，这是因为 BES 算法在进行分组时，对元组中的所有敏感属性均作为一个元组边，算法分组时主要是针对元组边进行操作，从而避免了敏感属性维数对算法执行效率的影响。

表 4-3-3 BES 算法执行时间随着敏感属性维数 d 的变化

敏感属性维数	2	3	4	5
BES 运行时间（ms）	17	15	19	17

4.4 本章小结

本章主要是针对一般多敏感属性数据发布满足复合敏感属性 L-多样性提出了一种新的分组方案：基于类二部图的边选择分组算法，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，最后通过在相同数据集情况下，通过与基于多维桶的 MBF 算法在数据隐匿率和附加信息损失度上验证了算法良好的分组性能，并通过对不同维数对算法执行时间的影响得出 BES 算法运行时间不受发布数据敏感属性维度的影响，证明 BES 算法运行时间的高效性。

第五章 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[27]。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，α)-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 $T=\{A_1, A_2, A_3, \bullet\bullet\bullet, A_p, S_1 S_2 S_3, \bullet\bullet\bullet, S_d\}$，其中 $A_i \in \{A_1, A_2, \bullet\bullet\bullet, A_p\}$（$1 \le i \le p$）为准标识属性，$S_j \in \{S_1, S_2, \bullet\bullet\bullet, S_d\}$（$1 \le j \le d$）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 t_i（$1 \le i \le n$）。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（w）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 $S_j \in \{S_1, S_2, \bullet\bullet\bullet, S_d\}$（$1 \le j \le d$）的值域 Dom（$S_j$）中，Dom（$S_j$）$=\{V_1^{Sj}, V_2^{Sj}, \bullet\bullet\bullet, V_n^{Sj}\}$，对敏感属性值 $V_i^{Sj} \in Dom(Si)$（$1 \le i \le n$ 具有其对应的敏感度权值，记为$VWeight(V_i^{Sj})$，且对$\forall V$，有$0 \le VWeight(V_i^{Sj}) \le 1$。

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 $V_i^{Sj} \le V_j^{Sj}$（$1 \le i \le j \le n$）当且仅当在同一敏感属性 S_j 中，敏感值 V_i 的敏感度权值低于敏感值 V_j 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 $SA=\{S_1, S_2, \bullet\bullet\bullet, S_d\}$，对于敏感属性 $S_j \in SA$，设定个性化的敏感度权值，表示为 $SWeight(S_j)$，且 $0 \le SWeight(S_j) \le 1$。

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 $t_i \in T$，对应元组边为 $t_i.E$。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，t_i 元组边权值与其 t_i 上的敏感属性值权值$VWeight(t[S_j])$和敏感属性权值 $SWeight(S_j)$的关系为：

$$TWeight(ti.E) = \sum_{j=1}^{n} (VWeight(t[Sj]) \times SWeight(Sj))$$

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敏感度权值。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）=0.7 × 0.3+0.2 × 0.7=0.35, TWeight（t2.E）=0.7×0.3+0.6×0.7=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，α)-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，α）- diversity 个性α化匿名模型。（L，α）- 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'={G_1, G_2, G_3, •••, G_m}，G_i（1≤i≤m）为 T'上的一个有效分组，若 G_i 满足：

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

（2）G_i 中所有元组对应的元组边权值之和小于等于α。即有 $\sum_{i=1}^{|G_i|} TWeight(ti) \leq \alpha$。其中 $t_i \in G_i$。若元组边权值之和表示为分组边权值 GWeight(G_i),则有$\mathbf{GWeight(Gi)} \leq \alpha$

则称分组 G_i 为满足（L，α）- diversity 个性化分组。若对 T'中所有分组均满足（L，α）- diversity 个性化分组，则称 T'是满足（L，α）- diversity 个性化隐私数据发布模型，其中α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，α）- diversity 个性化数据发布模型的发布数据是隐私安全的，因为（L，α）- 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，α）- 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[27]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α，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 S_j 平均权值表示为：

$$\overline{\mathrm{Weight(Sj)}} = (\sum_{i=1}^{|S_j|} VWeight(V_i^{Sj})) / |Sj| \quad (1)$$

其中 S_j 表示数据表 T 中第 j 个敏感属性，V_i^{Sj} 表示 S_j 中的第 i 个敏感值。例如 S_j 可以是表 4-2-1 中的敏感属性“Disease”，V_i^{Sj} 可以是“Disease”中的敏感属性值“Flu”。则 $\overline{\mathrm{Weight(Disease)}}$ =(0.2+0.6+0.5+0.9+0.9)/5=0.62。其中|S(Disease)|=5。

最大敏感度阈值：

$$\alpha = (\sum_{j=1}^{d} (\overline{\mathrm{Weight(Sj)}} \times SWeight(Sj)) \times L \times \beta \quad (2)$$

其中 L 和$\overline{\mathrm{(Weight(Sj)}}$的取值已经确定，$SWeight(Sj)$代表 S_j 的敏感属性权值，也

可由类似表 4-2-2 数据计算出来。β我们定义为（L，α）- diversity 个性化数据发布模型中的个性化可变参数。

可变参数β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，β基准参数为 1，其与α大小成正相关。β的值直接影响模型定义中α的值，β值越大导致α越大，α越大就对模型分组满足（L，α）- 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β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β取值过小，α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，α）- 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，α）- diversity 个性化数据发布模型通过元组敏感度最大阈值α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β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α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β来得到最佳取值 α，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，α）- 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α，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{A_1，A_2，A_3，•••，A_p，S_1，S_2，•••，S_d}，多样性参数 L

输出：准标识属性表 QIT，敏感属性表 ST

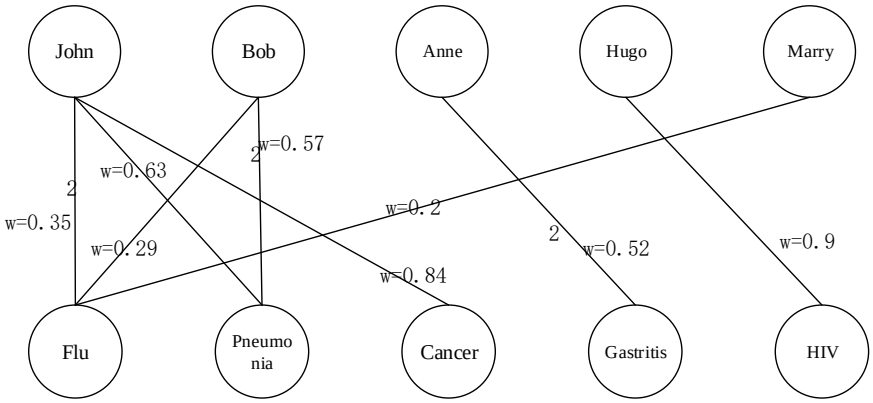
步骤：

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
4. while WTE 中的元组边不为空

5. 遍历 WTE-> t[i ••• n].E
6. if t_i.E 能在当前分组 G 上作不相交边选择且 t_i.E 不是排斥元组边
7. 将 t_i 将入到当前分组 G
8. if 当前分组 G 中元组数等于 L
9. 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
10. 停止遍历，重新进入到 while 循环
11. if 当前分组 G 中元组数不足 L
12. 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 t_i 加入到 GS_j 中 GS_j 依然满足多敏感属性（L，α）-diversity 个性化分组
16. 元组 t_i 加入到 GS_j 中，并将 t_i 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 4-2-3 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。

图 5-2-1 带权类二部图



令β = 1.1，则根据表 5-1-1 敏感度权值的设置和 5-1-2 中式（2）计算α得到α = 1.98，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（3，1.98）-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（3，1.98）-diversity 个性化的分组 G1={t1，t5，t6}，并且 GWeight（G1）=1.44≤ α。算法继续循环，最后可得另外两个分组 G2={t2，t4，t7}，GWeight（G2）=1.44≤ α，G3={t3，t8，t9}，GWeight（G3）=1.94≤ α，剩余元组{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 （3，1.98）-diversity 个性化敏感数据发布表

QIT(准标识属性表)

Tuple ID	QIDs	Group ID
t1	•••	G_1
t2	•••	G_2
t3	•••	G_3
t4	•••	G_2
t5	•••	G_1
t6	•••	G_1
t7	•••	G_2
t8	•••	G_3
t9	•••	G_3

ST(敏感属性表)

Group ID	Sensitive Attribute
G_1	<John，Flu>
	<Bob，Pneumonia>
	<Anne，Gastritis>
G_2	<John，Pneumonia>
	<Bob，Flu>
	<Anne，Gastritis>
G_3	<John，Cancer>
	<Hugo，HIV>
	<Marry，Flu>

5.3 L-拆分带权元组边选择分组算法—L-SWES

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，α）-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 同 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分组，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α的条件，从而造成虽然得到的分组虽然满足（L，α）-diversity 个性化匿名模型分组，但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，α）-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)

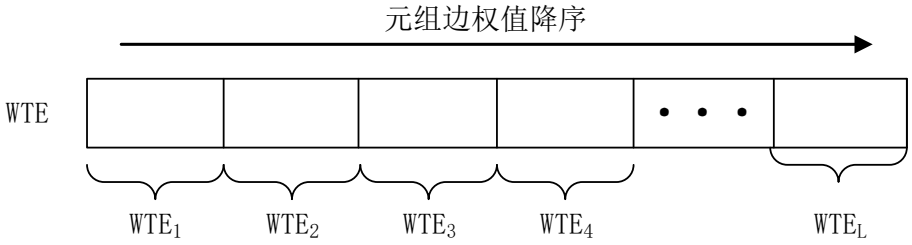
5.3.1 算法基本思想

WBES 算法中存在的主要问题是元组的分布问题，元组边由元组生成，元组边集合中的权值分布随机，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={WTE_1，WTE_2，WTE_3，•••，WTE_L}。$WTE_i(1\leq i\leq L)$称为一个等级子集合。并且有 $WTE=\bigcup_{i=1}^{L} WTE_i$。各子集合中的元组边敏感度权值依次递减，表示为：$WTE_1\geq WTE_2$，$\geq WTE_3$，$\geq$ •••，$\geq WTE_L$

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



WTE_Subset 中前 L-1 个子集合的大小相同为$\lfloor |WTE| / L \rfloor$，最后一个子集合的大小$L \leq |\ WTE_L\ | \leq 2L$。L-拆分元组边选择分组算法（L-SWES）在进行分组时，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集合中继续选取直到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{A_1，A_2，A_3，•••，A_p，S_1，S_2，•••，S_d}，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
12. 重新进入 while 循环
13. if 当前分组中元组数不足 L
14. 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
15. if WTE 中的元组数量小于 L
16. 将 WTE 中所有剩余元组全部加入到 RT 中，清空 WTE 中元组
17. 遍历 RT
18. for G[j ••• |GS|]
19. if t_i 加入到当前分组 G_j 中依然满足（L，α）-diversity 个性化分组
20. 元组 t_i 加入到 GS_j 中，并将 t_i 从剩余元组集合 RT 中移除
21. 隐匿 RT 中所有剩余的元组
22. 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，β依然取 1.1，则（L，α）-diversity 个性化数据发布模型中α＝1.98，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	WTE_i
t8	<Hugo，HIV>	0.9	
t3	<John，Cancer>	0.84	WTE_1
t2	<John，Pneumonia>	0.63	
t5	<Bob，Pneumonia>	0.57	
t11	<Bob，Pneumonia>	0.57	WTE_2
t6	<Anne，Gastritis>	0.52	
t7	<Anne，Gastritis>	0.52	
t1	<John，Flu>	0.35	WTE_3
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 WTE_1 中 t8 加入到当前分组 G 中，然后跳到 WTE_2 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 WTE_3 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≤α，不能满足（L，α）-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。

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导致敏感信息倾斜的问题，保

护了数据隐私。

表 5-3-2 L-WSES 分组算法得到最终数据发布表

QIT(准标识属性表)

Tuple ID	QIDs	Group ID
t1	•••	G_1
t2	•••	G_3
t3	•••	G_2
t4	•••	G_3
t5	•••	G_1
t6	•••	G_3
t7	•••	G_2
t8	•••	G_1
t11	•••	G_2

ST(敏感属性表)

Group ID	Sensitive Attribute
G_1	<Hugo, HIV>
	<Bob, Pneumonia>
	<John, Flu>
G_2	<John, Cancer>
	<Anne, Gastritis>
	<Bob, Pneumonia>
G_3	<John, Pneumonia >
	< Bob, Flu >
	<Anne, Gastritis>

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

```
# occupation
Tech-support:0.6
Craft-repair:0.5
Other-service:0.5
Sales:0.5
Exec-managerial:0.5
Prof-specialty:0.5
Handlers-cleaners:0.6
Machine-op-inspct:0.6
Adm-clerical:0.5
Farming-fishing:0.6
Transport-moving:0.5
Priv-house-serv:0.7
Protective-serv:0.7
Armed-Forces:0.8

# education
Bachelors:0.3
Some-college:0.2
11th:0.7
HS-grad:0.1
Prof-school:0.7
Assoc-acdm:0.4
Assoc-voc:0.4
9th:0.7
7th-8th:0.7
12th:0.7
Masters:0.5
1st-4th:0.7
10th:0.7
Doctorate:0.7
5th-6th:0.7
Preschool:0.8
```

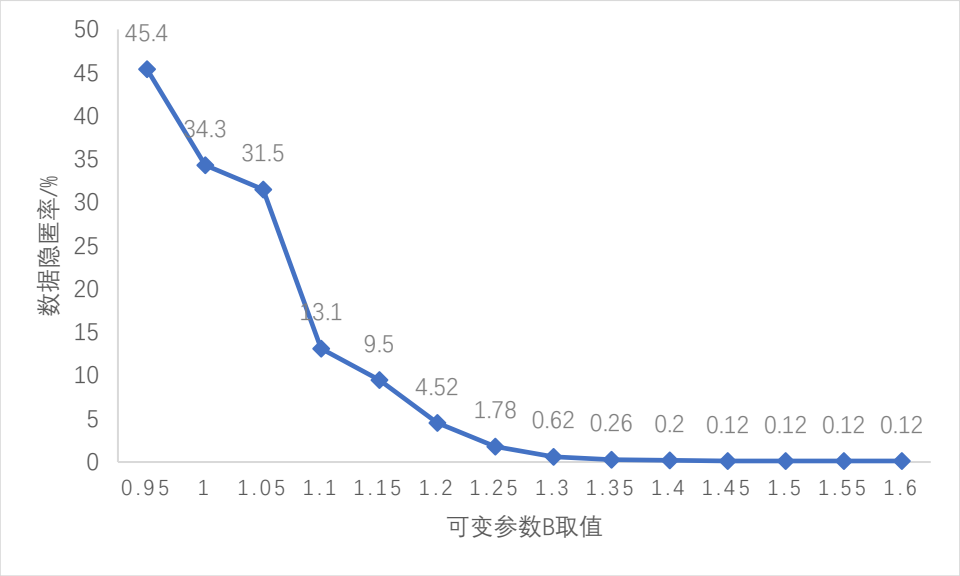
```
# marital-status
Married-civ-spouse:0.3
Divorced:0.7
Never-married:0.5
Separated:0.7
Widowed:0.8
Married-spouse-absent:0.8
Married-AF-spouse:0.8

# work class
Private:0.2
Self-emp-not-inc:0.6
Self-emp-inc:0.6
Federal-gov:0.7
Local-gov:0.6
State-gov:0.6
Without-pay:0.8
Never-worked:0.5

# race
White:0.2
Asian-Pac-Islander:0.5
Amer-Indian-Eskim:0.5
Other:0.5
Black:0.5
```

5.4.2 实验及结果分析

L-拆分带权元组边选择算法在进行分组时，由于可变参数β的取值不同，直接影响（L，α）- diversity 个性化匿名模型中分组敏感度阈值的大小，从而影响分组算法得到的有效分组数，图 5-4-1 给出了数据量 n=5k，敏感属性维数 d=3，多样性参数 L=3 时，不同β取值情况下，数据隐匿率的变化情况。可以看到随着β取值的逐渐增大，L-SWES 分组算法得到的发布数据隐匿率呈下降趋势，这是因

为β取值与模型定义中分组敏感度阈值成正相关，β值越大，分组敏感度阈值越大，非排斥元组边的选择空间更大，得到的有效分组就越多。的当β取值大于等于 1.4 时，数据隐匿率均为 0.12%，这是因为此时分组敏感阈值α的取值已经不再限制排斥元组边的操作，L-SWES 分组算法失去个性化发布性质，蜕变为一般 BES 分组算法。

图 5-4-1 可变参数 β 取值对数据发布隐匿率的影响（n=5k,L=3,d=3）



通过 5-4-1 实验结果，针对多敏感属性的（L，α）- diversity 个性化匿名模型，选取β=1.35 时能较好实现匿名效果和对高敏感度元组记录的保护。在此基础上我们对基于桶的分组算法 MBF，基于类二部图的元组边选择算法 BES，基于带权类二部图的 L 拆分元组边选择算法 L-SWES 在数据隐匿率和附加信息损失度等方面综合评定算法的性能。

图 5-4-2 为敏感属性维数 d=3，多样性参数 L=3 时，三种算法在数据隐匿率上的变化情况，可以看到各算法随着数据量的增大，数据隐匿率都呈下降趋势，原因与本文第四章中实验情况相同，个性化的分组算法 L-SWES 的数据隐匿率略微高于一般的分组算法 BES 数据隐匿率，这是因为加权的分组算法在考虑到需要满足（L，α）- diversity 匿名模型，分组时存在分组敏感度阈值的限制。单纯从分组的情形下看加权个性化分组得到的分组效果不如一般分组算法，但是，由于加权个性化分组算法考虑了分组时元组个性化的要求，得到的分组具有更高的可用性和保护性，且在可变参数β取值合理的情况下，加权个性化分组算法 L-SWES 依然拥有较好的实用性能。图 5-4-3 为分组算法的附加信息损失度随着数量的变化情况，这里加权个性化分组算法 L-SWES 的附加信息损失度虽然呈下降趋势但是低于一般分组算法 BES，这是因为加权个性化分组算法得到的有效分组

少于 BES 算法，且由于存在分组敏感度阈值的限制，在处理剩余记录的时候能够满足（L，α）- diversity 匿名模型的剩余记录较少，从而导致附加信息损失度的降低。

图 5-4-2 数据隐匿率随着数据量的变化（L=3，d=3）

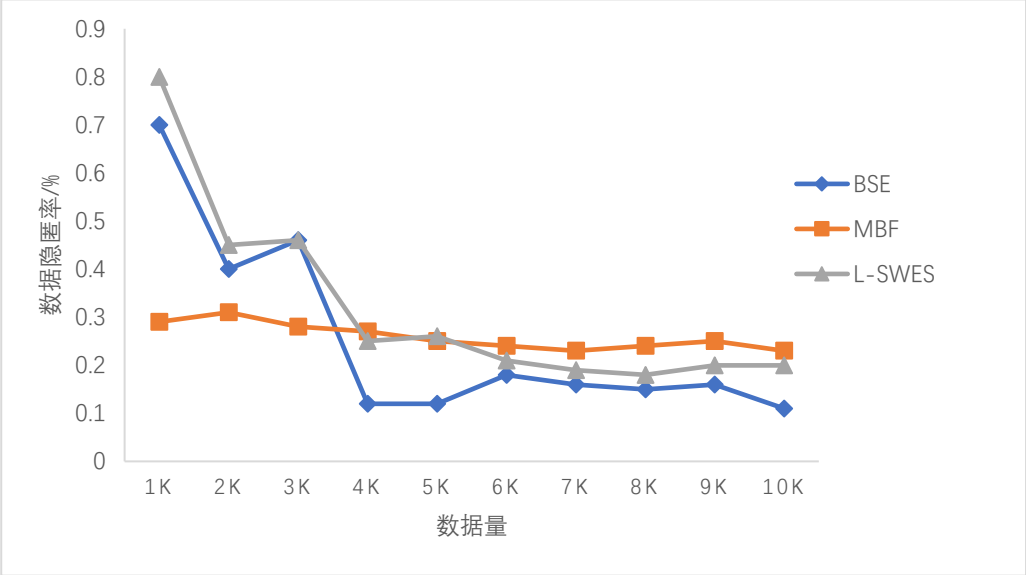


图 5-4-3 附加信息损失度随着数据量的变化（L=3，d=3）

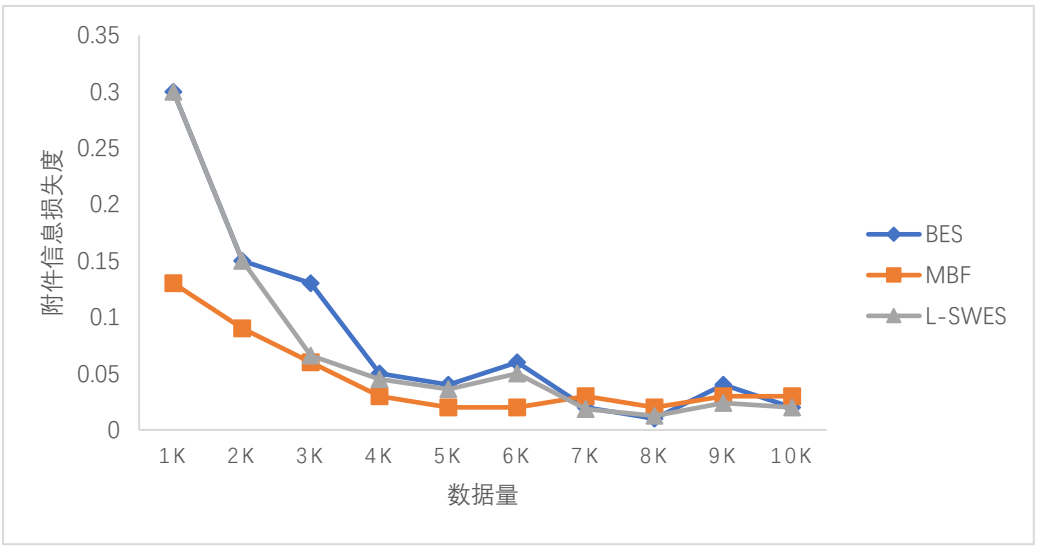


图 5-4-5 为数据量 n=5K，敏感属性维数 d=3 时，算法数据隐匿率随着多样性参数 L 的变化情况，可以看到图中三个算法随着 L 值的增大，数据隐匿率都呈现出比较明显的上升趋势，是因为随着多样性参数 L 的增大，对分组的要求的满足 L 多样性就需要敏感属性存在足够大的值域个数，实验数据中第三维敏感属性

值域个数最少的是 Marital-status，值域个数为 7，L 越接近这个值，在这一维上保证分组的 L-多样性越困难。

图 5-4-5 数据隐匿率随着多样性参数 L 的变化（n=5k, d=3）

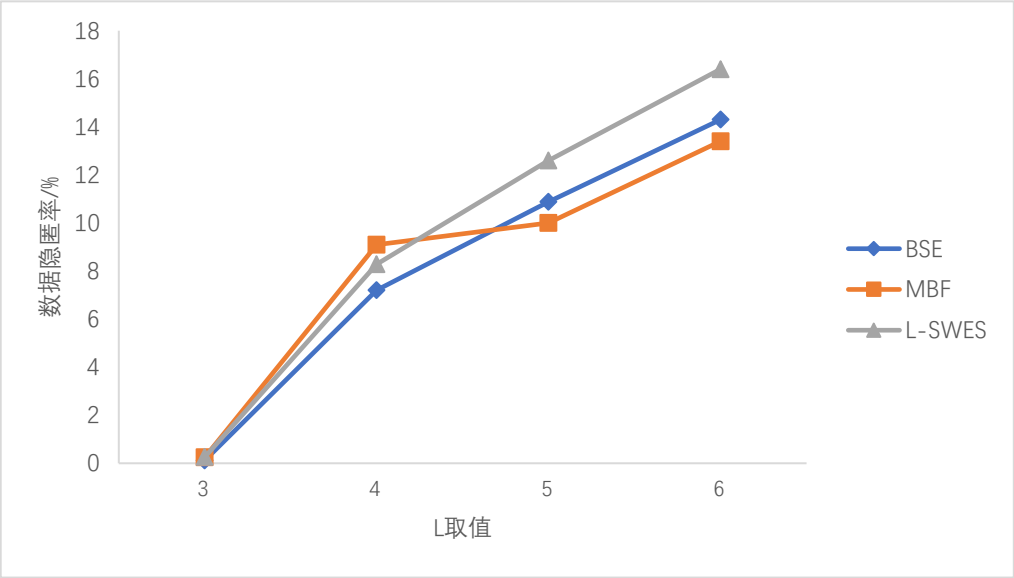


表 5-4-2 不同敏感属性维数下数据隐匿率的变化（n=5K，L=3）

	2	3	4	5
BES	0.12%	0.12%	0.23%	45%
L-SWES	0.12%	1.3%	4.6%	64%

表 5-4-2 为不同敏感属性维数对算法数据隐匿率的影响，数据量 n=5K，多样行参数 L=3，可以看出加权个性化分组算法 L-SWES 的数据隐匿率随着敏感属性维度的增大，数据隐匿率都要高于一般分组算法 BES，这是因为个性化分组算法在敏感属性个数增多时，分组满足（L，α）- diversity 个性化匿名模型相对困难，而一般分组算法 BES 没有这一约束。另外从表中可以看出当敏感属性维数达到 5 时，数据隐匿率陡然升高，这是因为数据集中，第五维敏感属性 Race 的取值值域个数为 5，造成在这一维上满足 5-覆盖性比较困难，从而导致数据隐匿率的大幅上升。我们可以看到，在隐私数据敏感属性的值域取值越小，对发布数据敏感属性的保护越困难。

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的基础上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个性化的分组算法，并对带权分组算法提出改进得到基于带权类二部图的元组边选择分组算法，并通过大量实验验证对比改进后算法的性能，最终证明改进后的算法得到比较好的多敏感数据个性化发布隐私保护效果。

第六章 总结与展望

6.1 研究工作总结

本文主要对数据发布中面向多敏感属性的个性化隐私保护技术进行研究。以往学术研究中对单敏感属性的隐私保护技术研究在应用于实际数据发布中时，会存在隐私泄露的问题，所以，随着研究的深入，关于数据发布隐私保护技术从单一敏感属性扩展到了对多敏感隐私属性的隐私保护技术的研究，并且根据实际应用场景对隐私保护技术提出了更高的要求--数据发布的个性化隐私保护。本文从实际场景出发，首先对国内外已有的个性化隐私保护技术进行深入研究，对现有个性化匿名模型存在的缺陷进行了详细分析，并提出一种新的个性化数据发布隐私保护模型，并实现其分组算法且对算法做出改进。综上，本文所做的主要工作如下：

（1）对多敏感属性场景下，提出一种新的数据发布分组算法--基于类二部图边选择分组算法（BES）。在一般性数据发布隐私保护模型中，重点分析了杨晓春等人提出的基于多维桶分组算法，并提出了一种新的基于类二部图的分组算法，根据敏感属性集映射到类二部图的不同点集合的方式得到对应的元组边，再由类似二部图寻找匹配的算法得到满足 l-多样性的分组。并通过算法的时间效率，数据隐匿率和附加信息损失度等方面证明了算法的高效性和可行性。

（2）提出一种新的个性化数据发布隐私保护模型--（L，α）-diversity 个性化数据发布模型。该模型通过设置调节个性化参数β从而得到合适的模型中分组最大阈值参数α，在分组满足多敏感属性 L-多样性的前提下保证分组的最大阈值不超过α，从而限制单个分组中高敏感度的元组出现频率，避免了高敏感属性值在分组中倾斜。且通过实验证明该模型的有效性。

（3）对以上提出的（L，α）-diversity 个性化数据发布模型设计一个基于带权二部图边选择的算法（WBES）该算法是在对一般性数据发布隐私保护中算法 BES 的加权改进。并进一步改进得到 L-拆分元组边选择分组算法(L-SWES)，改进后的算法在数据隐匿率和隐私保护效果有明显提升，并通过实验验证了 L-SWES 算法的有效性和可行性。

6.2 展望

本文虽然在针对多敏感属性个性化隐私保护提出了（L，α）-diversity 个性化数据发布模型并给除了相关算法实现，但研究依然有很多不足，未来研究方向可在以下几个方面展开：

（1）（L，α）-diversity 个性化数据发布模型依赖个性化参数β的设置，在特定条件下如何快速准确定义个性化的值，关于个性化参数β的定义与取值，可专门做研究，从而更加完善（L，α）-diversity 个性化数据发布模型。

（2）关于多敏感属性个性化发布个敏感属性值个性化指定目前只能靠数据发布者实际给出值，如何最大限度减小主观性给个性化数据发布中隐私保护产生的影响降低也可作为未来数据发布隐私保护研究的方向。

（3）本文工作主要是针对结构化数据集（或称关系型数据集）中的数据发布隐私保护问题，另外类似社交网络中等环境下产生的具有图结构的数据和本文研究的结构化数据有很大的不同，其匿名保护技术和相关算法也是具有巨大的研究价值，所以下一步的研究可以在非结构化数据的隐私保护和对应个性化发布上进行。

参考文献

[1] V. S. Iyengar. Transforming data to satisfy privacy constraints. in: the 8th ACM
SIGKDD International Conference on Knowledge Discovery and Data Mining. Edmonton, Alberta, Canada: ACM Press Press, 2002. 279~288

[2] Agrawal R,Srikant R.Privacy-preserving data mining[C].ACM SIGMOD.ACM Press,May 2000:P.439-450.

[3] Wang Y,Wu X,Wu L.Differential Privacy Preserving Spectral Graph Analysis[J]. Advances in
Knowledge Discovery and Data Mining.Springer Berlin Heidelberg,2013:329-340.

[4] 孙美丽.美国和欧盟的数据隐私保护策略[J].情报科学,2004,22(10):1265-1267.

[5] Fung B C,Wang Ke,Chen Rui,et al. Privacy-preserving data publishing:a survey on recent
developments[J]. ACM Computer Surveys(CSUR),2012,42(4):1-53.

[6] Sweeney L. k-anonymity: a model for protecting privacy[J]. International Journal of
Uncertainty,Fuzziness and Knowledge Based Systems,2002,10(5):557-570.

[7] Agrawal R,Evfimievski A,Srikant R. Information sharing across private database[C].ACM SIGMOD
2003.2003:P.86-97.

[8] R. Bayardo and R. Agrawal. Data privacy through optimal k-anonymization. in: the
21st International Conference on Data Engineering (ICDE’05). Tokyo, Japan: IEEE
Cpmputer Society Press Press, 2005. 217~228

[9] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan. Incognito: Efficient full-domain
k-anonymity. in: the 24th ACM SIGMOD International Conference on Management
of Data (SIGMOD’05). Baltimore, Maryland: ACM Press Press, 2005. 49~60

[10] P. Samarati and L. Sweeney. Protecting privacy when disclosing information:
k-anonymity and its enforcement through generalization and suppression. in: IEEE
Symposium on Research in Security and Privacy, May 1998.

[11] L. H. Cox. Suppression, methodology and statistical disclosure

control. Journal of
the American Statistical Association, 1980, 75(370):377~385

[12] R. Agrawal and R. Srikant. Privacy-preserving data mining. in: the 19th ACM
SIGMOD International Conference on Management of Data (SIGMOD' 00). Dallas,
Texas, USA: ACM Press Press, 2000. 439~450

[13] S. Chawla, C. Dwork, and F. McSherry et al. Toward privacy in public databases. in:
the 2nd Theory of Cryptography Conference (TCC' 05). Cambridge, MA, USA.2005. 363~385

[14] A. Machanavajjhala, J. Gehrke, and D. Kifer. l -diversity: Privacy beyond
k-anonymity. in: 22nd International Conference on Data Engineering (ICDE' 06).
Atlanta, Georgia, USA: IEEE Computer Society Press Press, 2006. 24

[15] A. Meyerson, R. Williams. On the complexity of optimal k-anonymity. in: the 23rd
ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS' 04). Paris, France: ACM Press Press, 2004. 223~228

[16] Li Ninghui,Li Tiancheng. t-closeness: privacy beyond k-anonymity and l-diversity[C]//Jarke M,Carey M J,Dittrich K R,et al. Proceedings of the 23rd International Conference on Data Engineering.Istanbul: IEEE,2007:106-115.

[17] 刘善成, 金华, 鞠时光. 数据发布中面向多敏感属性的隐私保护技术[J]. 计算机应用研究,2011,28(6):2206-2214.

[18] Ye X J,Zhang Y W,Liu M.A Personalized (α,k)-Anonymity

Model[C]//Proceedings of the 9th International Conference on Web-Age Information Management(WAIM’08).2008:341-348.

[19] 黄玉蕾, 林青, 戴慧珺. 基于多敏感值的个性化隐私保护算法[J]. 计算机与数字工程,2016,9:1761-1800.

[20] Fung B C M,Wang K,Chen R,etcl.Privacy—preserving data publishing: a survey onrecent developments[J]. ACM Computing Surveys, ACM press, 20 1 0, 42(4).

[21] Aggarwal C. C. On k-anonymity and the curse ofdimensionality[C], In Proceedings ofthe 31 st International conference Oil Very large data bases(VLDB), 2005.

[22] Samarati P. Protecting respondents’identities in microdata release[C]. In Proc of the TKDE. 2001: 1 01 0-1027.

[23] Terrovitis M,Mamoulis N,Kalnis P. Local and global recording methods for anonymizing set-valueddata[J].International Journal on Very Large Data Bases,2011,20(2):83-106

[24] Soria-Comas,Jordi.Domingo-Ferrer,Josep. Probabilistic k-anonymity through microaggregation and data swapping[C].2012 IEEE International Conference on Fuzzy Systems,2012:8.

[25] G.T.Duncan and D.Lamber. Disclosure-limited data dissemination[J].Journal of the American Statistical Association,1986,81:10-28.

[26] 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

[27] 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.

[28] 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

[30] 金华, 刘善成, 鞠时光. 面向多敏感属性医疗数据发布的隐私保护技术[J]. 计算机科学, 2011, 38(12):171-177.

[31] Xiao Xiaokui,Tao Yufei. Personalized Privacy Preservation[C] / / Proceedings of ACM SIGMOD International Conferenceon on Management of Data.New York,USA,2006:229-240.

[32] 韩建民, 于娟, 虞慧群, 贾洞. 面向敏感值的个性化隐私保护[J]，电子学报,2010,38(7):1723-1728P.

[33] 杨静, 王波. 一种基于最小选择度优先的多敏感属性个性化 1-多样性算法[J]. 计算机研究与发展, 2012, 49(12):2603-2610.

[34] 龚奇源, 杨明, 罗军舟. 面向关系-事务数据的数据匿名方法[J]. 软件学报, 2016, 27(11):2828-2842.

[35] Qiyuan Gong，Ming Yang，Zhenguo Chen，Junzhou Luo. Utility Enhanced Anonymization for Incomplete Microdata

[36] Sweeney L . K—anonymity : A model for protecting privacy. International Journal on Uncertainty，Fuzziness，and Knowledge-Based Systems，2002，10(5)：557—570

[37] Samarati P，Sweeney L. Generalizing data to provide anonymity when disclosing information

致谢

时间飞逝，三年宝贵的硕士生涯即将走完。在这三年间，感谢遇到那么多优秀的老师、同学、朋友，你们不仅让我学到了很多宝贵的知识，也让我在研究生期间过得充实而又意义非凡。

在此，最需要感谢的是我的导师叶春晓教授，叶春晓老师为人随和热情，对学生的学术研究也悉心指导，并定期和我们交流研究学习心得，给出学术研究上的意见和建议。在我研究生期间叶春晓老师为我指明了研究方向，时常询问并关心我的学术研究进展，对我起到了良好的敦促作用，再次感谢叶春晓老师，感谢您这几年的悉心指导与辛勤的付出！

其次，感谢计算机学院辛苦耕耘的老师们，你们学识渊博，治学严谨，诲人不倦的精神不仅仅是帮助我增长了知识，更是开拓了我的视野，让我学会了去探索未知，这将使我终身受益！

感谢身边的每一位同学、朋友、师兄师姐们，我不仅仅在你们身上学到那么多优秀的品质，更是让我认识到自己的不足，感谢你们的优秀，让我时刻警醒自己积极向上。更是因为有了你们的陪伴，让我的研究生生涯像是一个大家庭，过得快乐而又充实！

最后，感谢我的父母和女友，感谢你们这么多年来对我的学业无论是物质还是精神上都给予最大的支持，使我在学习生活中坚定不移向前，感谢你们！

再次感谢所有关心我、帮助过我的人们。