数据发布中多敏感属性数据隐私保护

方法研究



重庆大学硕士学位论文

（专业学位）

学生姓名：

指导教师：

学位类别：工程硕士（计算机技术）

重庆大学计算机学院

二O一八年四月

**Privacy preserving method for multi sensitive attributes data in Data Publishing**

****

A Thesis Submitted to Chongqing University

in Partial Fulfillment of the Requirement for the

Professional Degree

**By**

**Supervised by**

**Specialty: ME (Computer Technology)**

College of Computer of

Chongqing University, Chongqing, China

April 2018

# 摘 要

在计算机网络技术快速发展的大背景下，互联网为了给人们带来越来越大的便利，大量收集用户的各方面信息。由于互联网的开放性，导致用户信息共享变得也变得越来越简，同时用户个人敏感信息发生泄漏的风险大大增加。为了进行数据挖掘，数据研究科研，在某些情况下数据拥有者需要在互联网上发布包含用户个人信息的数据集，这种情况下的信息共享更容易造成用户个人的隐私信息泄露。因此，在这些数据发布机构发布数据之前需要对原始数据集通过一定方式隐藏用户的身份保护敏感数据，数据发布中的隐私保护也成为隐私领域的主要研究内容，其主要目标是对发布数据中在保证用户信息的隐私不被泄露的同时尽可能地提高发布数据的可用性，实现高效安全的信息共享。

在数据发布的实际应用中，通常面临的情况是数据集包含多敏感隐私属性，并且针对敏感信息需要进行个性化保护，因此多敏感属性隐私保护及个性化发布是当前数据发布中的研究热点，本文重点分析了现有的基于多维桶技术和利用L-覆盖性聚类方法的多敏感属性数据发布方法，并提出了一种新的基于类二部图边选择的分组算法（BES），并通过实验证明了BES算法的有效性。

本文继续分析了在多敏感属性个性化数据发布的情况下，现有的加权多维桶分组算法（WMBF），最小选择度优先分组算法，完全（a,k）-anonymity模型等均存在敏感度高的元组划分到同一等价类分组中，造成隐私属性值倾斜，容易受到同质攻击的问题，并且制定个性化方案时只考虑了敏感属性值的敏感度而没有综合考虑敏感属性自身的敏感度问题。因此本文在综合考虑敏感值和敏感属性自身敏感度的基础上，为避免发布数据中同一分组隐私属性值倾斜，在L-多样化模型的基础上提出（L，）- diversity个性化匿名模型，针对该模型实现加权的基于类二部图边选分组算法（WBES）和其改进算法L-拆分元组边选择分组算法(L-SWES)。实验结果表明本文提出的算法能有效避免同质攻击，并且能够得到较好的数据发布效果。且分组算法在执行时间上不受敏感属性维度的影响，保持比较好的算法效率。

**关键词：**数据发布 隐私保护 多敏感属性 个性化 分组算法

# ABSTRACT

Under the background of rapid development of computer network technology, the Internet has collected a great deal of information from users in order to bring more and more convenience to people. Due to the openness of the Internet, the sharing of user information has become more and more easily, and the risk of leakage of sensitive personal information of users has greatly increased. In order to conduct data mining, data research and scientific research, in some cases, data owners need to publish data sets containing personal information of users on the Internet. In this case, personal privacy information of users is more likely leaked through the information sharing. Therefore, before these data issuing agencies publishing data, they need to hide the user's sensitive data from the original dataset in a certain way. The privacy protection in data publishing has also become the main research content in the privacy field. The main goal is to increase the availability of data in the published data while ensuring the privacy of user information is not leaked, to achieve efficient and secure information sharing.In the actual application of data publishing, it is often the case that the dataset contains multi sensitive privacy attributes and personalized protection for sensitive information. Therefore, multi sensitive attribute privacy protection and personalized publishing are hot topics in current data publishing research. This paper focuses on the existing multi sensitive bucket data technology and L- sensitive clustering method. A new grouping algorithm (BES) based on the Bigraph-similar Edges Selection is proposed, and the effectiveness of BES algorithm is proved by experiments.

In the actual application of data publishing, it is often the case that the dataset contains multi sensitive privacy attributes and personalized protection for sensitive information. Therefore, multi sensitive attribute privacy protection and personalized publishing are hot topics in current data publishing research. This paper focuses on the existing multi sensitive bucket data technology and L- sensitive clustering method. A new grouping algorithm (BES) based on the Bigraph-similar Edges Selection is proposed, and the effectiveness of BES algorithm is proved by experiments.

In this paper, we continue to analyze the existing weighted multi-dimensional bucket grouping algorithm (WMBF), the minimum selection priority grouping algorithm, the fully (a, k) -anonymity model, and so on, may to divide the tuples with high sensitivity to the same equivalence class，this situation will cause privacy attribute value tilt, vulnerable to homogeneous attacks. And the sensitivity of sensitive attribute values is only considered when making personalized programs, instead of considering the sensitivity of sensitive attributes themselves. Therefore, considering the sensitivity of sensitive value and sensitive attribute comprehensively, In order to avoid the tilt of the same group privacy attribute values in publishing data, we propose a (L, a) - diversity based on the L- diversity. Weighted Bigraph-similar Edges Selection algorithm (WBES) and its improved algorithm L-Split Weight Edge Selection grouping algorithm (L-SWES) are proposed for the implementation of the model. Experimental results show that the algorithm proposed in this paper can effectively avoid homogeneity attacks and get better data release effect. And the grouping algorithm is not affected by the dimension of sensitive attributes in execution time and maintains relatively good algorithm efficiency.

**Key Words:** Data publishing, Privacy protection, Multi sensitive attributes, Personalization, Grouping algorithm

# 目 录

[中文摘要 I](#_Toc511603011)

[英文摘要 II](#_Toc511603012)

[1 绪论 1](#_Toc511603014)

[**1.1 数据发布研究背景及意义** 1](#_Toc511603015)

[**1.2 数据发布中隐私保护研究现状** 2](#_Toc511603016)

[1.2.1 一般性数据发布研究 2](#_Toc511603017)

[1.2.2 个性化数据发布研究 3](#_Toc511603018)

[**1.3 本文主要研究内容与组织架构** 3](#_Toc511603019)

[2 数据发布中的隐私保护 5](#_Toc511603020)

[**2.1 隐私保护中的数据泛化方法** 5](#_Toc511603021)

[**2.2 隐私保护中的匿名模型** 7](#_Toc511603022)

[2.2.1 k-匿名模型 7](#_Toc511603023)

[2.2.2 L-diversity模型 8](#_Toc511603024)

[2.2.3 t-closeness模型 10](#_Toc511603025)

[**2.3 本章小结** 10](#_Toc511603026)

[3 多维敏感属性数据发布中的隐私泄露 11](#_Toc511603027)

[**3.1 多敏感属性数据发布中的问题研究分析** 11](#_Toc511603028)

[**3.2 多敏感隐私属性的数据发布方法** 12](#_Toc511603029)

[3.2.1 基于多维桶分组技术的隐私数据发布方法 13](#_Toc511603030)

[3.2.2 基于L-覆盖性聚类分组的隐私数据发布方法 15](#_Toc511603031)

[**3.3 多敏感属性的个性化隐私保护** 16](#_Toc511603032)

[3.3.1 面向多敏感属性的个性化数据发布算法 17](#_Toc511603033)

[**3.4 本章小结** 19](#_Toc511603034)

[4 面向多维敏感属性的数据发布 21](#_Toc511603035)

[**4.1 多敏感属性隐私数据发布问题** 21](#_Toc511603036)

[4.1.1 发布数据表中的属性定义 21](#_Toc511603037)

[4.1.2 有损连接发布 21](#_Toc511603038)

[**4.2 基于类二部图边选择的多敏感数据分组算法--BES** 22](#_Toc511603039)

[4.2.1 算法基本思想 23](#_Toc511603040)

[4.2.2算法描述 25](#_Toc511603041)

[4.2.3 BES算法实例应用 26](#_Toc511603042)

[**4.3 实验结果及分析** 28](#_Toc511603043)

[4.3.1 实验数据集 28](#_Toc511603044)

[4.3.2 实验及结果分析 29](#_Toc511603045)

[**4.4 本章小结** 31](#_Toc511603046)

[5 面向多敏感属性的个性化发布模型 33](#_Toc511603047)

[**5.1 多敏感属性(L，)-diversity 个性化数据发布模型** 33](#_Toc511603048)

[5.1.1 相关定义与描述 33](#_Toc511603049)

[5.1.2 (L，)-diversity 个性化数据发布模型描述 36](#_Toc511603050)

[**5.2 带权类二部图边选择分组算法--WBES** 37](#_Toc511603051)

[**5.3 L-拆分带权元组边选择分组算法—L-SWES** 39](#_Toc511603052)

[5.3.1 算法基本思想 40](#_Toc511603053)

[5.3.2 算法描述 41](#_Toc511603054)

[5.3.3 L-SWES算法实例 42](#_Toc511603055)

[**5.4 实验及结果分析** 43](#_Toc511603056)

[5.4.1 实验数据 43](#_Toc511603057)

[5.4.2 实验及结果分析 45](#_Toc511603058)

[**5.5 本章小结** 48](#_Toc511603059)

[6 总结与展望 49](#_Toc511603060)

[**6.1 研究工作总结** 49](#_Toc511603061)

[**6.2 展望** 49](#_Toc511603062)

[致 谢 51](#_Toc511603063)

[参考文献 52](#_Toc511603064)

# 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，社会的高度信息化使得网络中产生海量的各种数据，其中大量涉及个人的数据能够用数据计算的方法进行收集和分享。大量数据信息的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恶意使用，个人敏感信息不被泄露等问题有了极大的关注[1]。然而在数据管理与信息共享等实际应用中数据发布是非常关键的一部分。在大量的数据中通过算法搜索其中潜在的价值，分析海量数据提取出隐藏的、有价值的信息，对于商业科研等领域举足轻重。从大量数据中提取有价值的信息是否能取得好的效果取决于是否有可用的高质量数据。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存在比较多的机构需要定期对公众发布一些数据。例如，出于医学研究统计等原因，医院需要定期发布的患者医疗统计数据。在当今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，收集海量的数据、公布和各种对数据的分析变得越来越简单易行[2]，这同时也给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，可以找出各种疾病之间的关联性或分布区域性等特点，具有不可估量的科学研究价值。但在对医疗数据分析的过程中，不可避免地会涉及到患者的一些个人数据，从而可能导致患者包括疾病在内等敏感信息的泄露。因此，对于有效地解决带有敏感信息的数据发布中可能存在的隐私泄露问题，已经成为数据隐私安全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]。

互联网出现以前，用户个人信息在政府机构，银行或医院等可信任组织的保护下相对比较安全，互联网络出现后并快速发展，大量机构将收集到用户个人信息保存在互联网中。由于互联网络的开放特性，用户个人隐私信息泄露的风险大大增加。更是因为互联网络的开放特性，通过数据发布达到的数据共享将会承担更大的隐私泄露风险。在这样的大背景下，在数据发布中如何针对用户个人的敏感隐私信息进行有效保护，并尽量减小对共享数据质量的影响已经成为了一个新兴热门的研究方向[5-7]。在互联网海量的数据中，数据共享将带来巨大的商业科研价值，但这一过程同时也给隐私保护问题带来巨大挑战。如果不考虑用户隐私问题而进行数据发布共享，则可能给用户自身或者社会造成严重的损失。因此，在对数据发布中的隐私保护进行研究时，重点是解决数据共享中个人隐私信息保护的问题，完善隐私保护在数据共享中的应用，使数据拥有者在进行数据共享、数据发布时能够更加高效、安全地进行，并且使得发布共享的数据具有较高的可用性以供社会团体，研究机构及个人研究分析。由此既保留了数据的实用价值，有可以最大限度地保护用户个人信息，降低隐私信息泄露的风险，从而解决数据共享发布过程中的诸多隐私保护相关问题。

## 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，数据基本上以表的方式存储。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。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集中的数据进行匿名处理。其中处理方式包括泛化[8-9]处理、压缩[10-11]处理、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是指身份泄露和敏感信息泄露。关于这两个概念，身份泄露是指当目标个体的身份能够在发布数据中与某条具体记录直接联系起来，从而相关信息被泄露，而敏感信息泄露是指发布数据中泄露个体记录的一些私人敏感信息。

近年来，在数据发布中的隐私保护主要是在三个方面进行，首先是对数据发布隐私保护模型的研究，这也是研究中的基础部分，目的是通过隐私保护模型来衡量最终发布数据的隐私保护有效性，判断满足此类模型的发布数据表能否提供足够力度的隐私保护效果。其次是根据提出的隐私保护模型研究其实现算法，也就是在证明隐私保护模型的有效性后如何通过某种高效算法来处理待发布数据得到满足模型的发布数据。最后是对满足隐私保护模型的待发布数据，通过何种方式可提高发布数据的可用性的研究。在这三个方面，模型的研究是第一步，对隐私保护模型的研究有最早提出的k-anonymity[6]和L−diversity[14]等隐私保护规则模型，也存在在此基础上的多种改进型隐私保护模型，如t-closeness隐私保护模型。对隐私保护模型实现算法的研究是与模型本身强相关的，提出一种数据发布隐私保护模型肯定需要实现满足该模型的分组算法，例如针对k-anonymity模型的匿名数据近似算法[15]。关于数据发布的可用性主要是在发布数据满足隐私模型的前提下通过何种方式发布能提高数据可用性，例如数据泛化或者Anatomy[29]方法等。在对数据发布隐私保护的三个部分的研究中，出现许多经典的匿名模型，如A. Machanavajjhalla 等人[14]在k-匿名模型的基础上进一步提出了L-多样化模型（L-diversity），LiNinghui 等[16]提出了t-接近模型（t-closeness），进而国内学者杨晓春等人提出了复合敏感属性L-多样性模型[30]，这些经典模型的提出受到了相关学者的关注与研究并提出了许多改进的模型和实现算法。

### 1.2.2 个性化数据发布研究

隐私保护中的个性化数据发布最早是由Xiao Xiaokui,Tao Yufei最早提出的[32]。个性化数据发布主要是针对发布数据集中的不同敏感属性指定不同的发布规则，由于发布数据中的包含的敏感信息存在敏感度高低的问题，通常情况下高敏感度的隐私信息更具有研究价值，而个性化数据发布可针对不同情况下的场景需求，针对不同敏感属性指定不同的发布规则，从而让个性化数据发布更具有实际应用价值，所以，在个性化数据发布概念提出后，众多学者在个性化隐私保护技术上进行了大量的研究。例如，Ye等人在已有的（a，k）-anonymity 个性模型的基础上进行研究和改进提出泛化敏感属性值的个性化数据发布方法[21]，还有韩建明等人对不同敏感属性进行权值划分，将敏感属性值的敏感度分级，提出一种完全（a，k）-anonymity 模型[31]。黄玉蕾等人在考虑敏感属性值敏感度的基础上结合敏感属性自身敏感度，对k-匿名模型进行改进得到个性化的匿名模型[22]，并在此基础上实现了基于敏感值的个性化隐私保护算法。

## 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，介绍了数据发布隐私保护研究中主要采取哪些方法，并给出常用的数据发布模型和隐私保护中常用技术，分析了这些模型解决了哪些隐私保护中存在的问题和各模型仍然存在的缺陷，最后阐明当前数据发布隐私保护研究领域研究的发展方向。

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。

# 2 数据发布中的隐私保护

数据发布中的隐私保护研究的主要目的是保证个人的隐私信息不被泄露的同时尽可能地提高发布数据的可用性。隐私保护数据发布场景中通常分为数据提供者、数据收集者、数据发布者、数据接收者等四种角色。通常情况下数据的发布者也是数据收集方，一般是值得信任机构，但是在开放网络环境下我们无法确定数据接收者的具体身份，并且对于数据接收者将如何使用这些发布出来的数据也无不可能全部知晓。所以，为了尽可能保护数据中的用户隐私信息，只能够在数据发布过程中对原始数据进行尽肯能的处理，对发布过程中的处理方法进行研究，从而得到安全的、高质量的最终发布数据。

## 2.1 隐私保护中的数据泛化方法

匿名化是对数据表中原有的数据进行处理，通过这种方式来隐藏数据表中个体身份或敏感信息，从而达到保护个体隐私信息的目的，其中处理数据的方法称为匿名化方法。现有研究常用的匿名化方法主要有泛化、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

Table 2.1 Raw data table

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 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首次提出，是实现匿名化模型的典型方法。泛化操作就是将数据集中记录的精确取值变成范围的、模糊的取值。准标识属性的取值包括分类型和数值型，数值型属性通常被泛化成区间，分类型属性通常用一个更一般的值来取代。例如：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\*\*)。对表2-1可进行如下泛化，得到泛化后的数据表如表2.2。

表2.2 准标识属性泛化表

Table 2.2 Quasi identity generalized table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 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 |

通过泛化处理的数据对于原始数据来说值准标识属性上含有更少的信息，这样一方面能够较好地保持原有数据的统计特性，另一方面又能保证数据的可用性，当然实际发布中会为隐匿类似“Name”这样的标识属性。泛化是对一些属性值进行扩大化或区间化的模糊处理，经过这样的操作会使匿名表中的信息含量将会降低。对于泛化方法，Fung等人总结了全域泛化模式、兄弟泛化模式、子树泛化模式、多维的泛化和单元泛化模式这五种泛化模式[29]。数据泛化容易受到“维度灾难”影响。Aggarwal等人[21]的研究指出，当数据维度达到l0至15维时，泛化算法将会出现丢失所有信息的情况。为了避免此类问题，Xiao等人提出了基于有损分解的数据匿名操作Anatomy[29]。

② 分解排列（Anatomy）

分解操作与泛化和隐匿相比，并不修改准标识属性值或敏感属性值，Anatomy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内容的前提下，保证不泄露用户的隐私信息。主要操作是：首先对敏感属性值进行分组，然后将敏感属性与其他属性分开发布，从而扰乱准标识符与敏感属性之间的对应关系。如图2-1所示：

图2.1 通过有损分解实现数据匿名

Fig 2.1 Data anonymity through lossy decomposition

原本准标识符qid与敏感属性sa之间是一一对应关系（图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（图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（图c）。在这种情况下，若果要重构经过Anatomy处理的数据就会出现很多不在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身份和敏感信息，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，但主要是针对敏感属性为数值型的待发布数据集，它首先将待发布数据记录划分为若干个分组，然后再在各分组中随机打乱敏感属性值的顺序，达到扰乱准标识符与数值型敏感属性的关联关系，从而达到隐私保护的效果。

## 2.2 隐私保护中的匿名模型

### 2.2.1 k-匿名模型

如果对数据发布表仅进行标识符隐匿（如隐匿掉姓名与身份ID），无法完全避免恶意用户利用外部知识与发布表进行链接确定发布数据表中的个体身份，并获取敏感隐私信息，这种方式被称为链接攻击。链接攻击是由于数据发布方不能确定数据接收方拥有什么样的背景知识，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的方法，从中获取隐私信息 [23]。为了避免发布的数据遭受链接攻击，由P.Samarati 和L.Sweeney 提出的k-匿名模型[24-25]，通过限制发布数据表中的等价分组满足k-匿名化是一种重要方法。k-匿名模型的主要思想是：对待发布数据表进行处理，使得最后发布的数据表中每个等价类分组中都存在至少k个在准标识属性上取值完全相同的个体记录。即使攻击者可以通过准标识符属性与的其他数据链接，也不能唯一地识别每个元组所有者的具体身份，只能以不超过1/k的概率识别元组的个人身份，通过这种方式来减少隐私泄露的风险。如表2.3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2.3 满足3-匿名模型的发布表

Table 2.3 Publishing table to satisfy the 3-anonymous model

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 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 |

k-匿名模型中k的取值是由用户定义的整型可变参数，对于k的大小，数据发布者可以通过调整其大小来达到不同的隐私保护程度。对数据进行满足k-匿名的处理后，肯定会产生一定程度的信息损失。因此最终发布数据相对于原始数据信息的损信失量可作匿名化数据质量的一种度量。在实际应用场景中，需要合理地选取k-匿名模型中k的值，k值越大，需要对数据表进行的泛化层次越高，发布数据的质量就会下降，k值越小，对隐私信息的保护力度越小，越容易造成隐私信息泄露。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，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

在k-匿名模型中通过对准标识属性进行泛化处理后得到等价类分组，仅仅考虑了对准标识属性的约束，缺乏对敏感属性的约束，造成k-匿名化后的数据依然可能遭受同质攻击和背景知识攻击，Machanavajjhala等人在k-匿名模型的基础上提出L-diversity匿名模型[14]。L-多样性与k-匿名类似，都是通过降低数据的表示粒度，隐匿敏感属性和准标识属性的对应关系来达到隐私保护的效果。这种对数据粒度的降低是一种折中方案，虽然会对数据的可用性造成一定的影响，但 也提高了发布数据的安全性。L-多样性是在k-匿名的基础上对其进行的扩展，k-匿名使用泛化和抑制降低数据表示的粒度，从而使等价类分组中至少有k条记录。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-多样性能保证不会受到同质攻击。

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，通过这种方式可以避免攻击者利用同质性攻击来得到敏感信息。例如，表2.4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2.4 满足3-diversity的发布数据表

Table 2.4 Publishing table to satisfy the 3-diversity model

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 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模型无法阻止概率推理攻击，如果存在一个等价类分组中的某些敏感属性值比其他敏感属性取值的频率高出许多，这使得攻击者能够得知组中的某一记录高可能性拥有该属性值。

③ L-diversity模型要求较高，它假设敏感属性的分布是相近的，当敏感属性分布不均匀时，难以得到足够的满足L-diversity的等价类分组。

### 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-邻近模型可以看作是对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L-多样性模型的基础上，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模型主要是针对L-diversity和k-k-anonymity中存在的同质攻击和同一等价类分组中存在大量敏感值相似的元组的问题提出了解决方案，Li Ninghui 在文献[16]中指出t-closeness 对属性泄露的保护，但是不包括对身份泄露的保护。t-closeness 主要不足体现在：

① t-colseness 主要是需要待发布数据隐私属性的均匀性，无法完全保证可以避免对k-anonymity 的背景知识攻击；

② t-colseness会导致发布数据可用性降低，因为它要求相同等价类中的敏感属性分布相同或相近，如果增大参数t的值，虽然可以保证发布数据的可用性，但是也会增大隐私泄露风险。

## 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-多样性模型和在此基础上改进的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各个模型存在的不足之处。

# 3 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

早期对数据发布中的隐私保护技术研究中，大多数敏感数据发布方法都是针对单一敏感属性的隐私保护。但是，在实际应用中，发布的数据集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 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得知一个主治医生专治哪几种疾病，其具体的关联性如表3.2 所示。

表3.1 原始医疗信息表

Table 3.1 Raw data table for medical information

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 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 |

表3.2 属性间关联表

Table 3.2 Inter attribute association table

|  |  |
| --- | --- |
| 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3.3 所示。由表3.3 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取表3.2内容的背景知识，若攻击者得知该个体的Physician 属性值“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-多样性原

表3.3 医疗发布数据3-diversity匿名表

Table 3.3 Medical publication data table that meets 3-diversity

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 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 |

则，造成隐私泄露的风险升高。可以看出因为敏感属性Physician和Disease 间存在一定的关联性，即使数据发布表中的敏感属性满足l-diversity 匿名模型，还是会存在隐私泄露的风险。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。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。

## 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，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]提出了数据发布中多敏感属性L-diversity 的概念，并对其进行了如下定义：

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SA中任意选取一个敏感属性，其余的准标识符属性和敏感属性均作为准标识符属性，若此时发布数据表T 均满足L-diversity，则说明发布数据表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[35]，通过这种方式可以较好解决多敏感属性中数据发布隐私保护的问题。但是这种方式也一个明显的缺点：当敏感属性维数增加时，等价类分组为了满足多敏感属性L-diversity 规则就必须搜索更多的记录，这必然会导致数据泛化程度升高，数据表示粒度降低，从而造成大量的信息损失，降低了发布数据的可用性。

### 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[30] 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[30]。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 =T，并且=（1ijm）。

④ 复合敏感属性L-多样性分组. 对于复合敏感属性的分组G，如果G中的每一维敏感属性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。

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。杨晓春等人为了找到具有多敏感属性的待发布数据表T的分组方案，在有损连接技术的基础上提出基于多维桶的多敏感属性数据发布分组技术，使最终得到的所有分组都满足复合敏感属性L-多样性。

多维桶的构造原理如下：复合敏感属性中的每个维度对应多维桶的一维，根据数据表中每条记录对应的复合敏感属性向量将其映射到对应的桶中。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 的原始数据为例，建立多维桶如表3-1-4 所示。

表 3.4 由表3.1构造的d维桶（d=2）

Table 3.4 D dimensional bucket constructed from table 3.1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 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，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

金华等人在对多维桶分组算法存在的问题进行研究分析后，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多敏感属性L-覆盖性的定义来进行分组算法，并给出相同敏感属性集和平均概率泄露度来进行算法操作，并对算法的分组效果进行评估。

① 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=。

② 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。

我们以表3.1为原始数据表为例，利用L-覆盖性聚类分组算法进行分组。令L=3，第一步将数据集T中的每一条记录都看作是一个初始分组，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3.5所示。

表 3.5 L-覆盖性聚类分组算法发布数据

Table 3.5 L- overlay clustering grouping algorithm for publishing data

|  |  |  |
| --- | --- | --- |
| 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，可能不满足复合敏感属性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]在简单（a,k）-匿名模型和一般（a,k）-匿名模型的基础上改进得到完全（a,k）-匿名模型，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，通过这种方式实现对不同敏感属性的值在同一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）-匿名约束仅仅是面向一个特定的敏感值的，待发布数据表T中经过匿名分组后得到最终发布数据表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）-匿名模型是在简单（a，k）-匿名模型的基础上从对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3.6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 3.6 完全（av，3）-匿名表

Table 3.6 Complete (a, 3) - anonymous table

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 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，不然很难生成满足完全（a，k）-匿名约束的发布数据表。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：

② 基于最小选择度优先的多敏感属性分组算法

杨静等人[34]提出了一种基于最小选择度优先的分组算法，实现敏感属性的个性化隐私保护需求，并且发布数据表满足多敏感属性L-多样性的要求。

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。

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。

# 4 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

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]分析了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。

关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，在复合敏感属性的数据集中得到的分组每一维敏感属性均满足L-多样性模型，达到隐私保护的要求。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三类：

① 标识符（Identifier，ID），数据表中能够唯一确定个体记录具体身份的属性，例如姓名，身份证号等。

② 准标识符（Quasi-Identifier，QID），数据表中组合起来可以确定个体记录身份的属性，如出生日期，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。

### 4.1.2 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 原始表

Table 4.1 Original table

|  |  |  |
| --- | --- | --- |
| 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，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两个表发布，两个表之间仅通过分组编号相连，如表4.2所示。

表 4.2 有损分解发布表

Table 4.2 Release table by Anatomy

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

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

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。

## 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={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[14]。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[30]。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 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>中点集V可划分为两个子集V1,V2，且满足V1∩V2 =，V1∪V2 = V，并且图BG中任意一条边上的两个点一个属于V1，另一个属于V2，则称G为二部图（Bigraph）。

针对待发布医疗数据表3.1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的各元组可得到如下图的二部图：



图 4.1 表3.1数据表敏感属映射的类二部图

Fig 4.1 Bigraph-similar mapped by the table 3.1

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类二部图（Bigraph-similar）：



图 4.2 3个敏感属性映射的类二部图

Fig 4.2 Bigraph-similar mapped by 3 sensitive attributes

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的元组边，记为ti.E。V(ti.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=。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，在不破坏L-多样性的基础上，分组越小则数据的利用度越高，信息损失也越小。理想的情形是若得到的分组大小刚好为L，并满足多敏感属性L-多样性，也就是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，可能存在某些分组的记录条数会超过L，虽然满足L-多样性但会造成附加的有损连接信息损失。这里采用文献[17]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，然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。最后从最后发布的数据中隐匿掉不包含在任何分组的个体记录。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，最好的情况是隐匿率SuppRatio=0。本文将根据数据隐匿率和附加信息损失度一起来衡量发布数据的质量。

### 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加入到当前分组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4.3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4.3。

表 4.3 待发布数据表

Table 4.3 Raw data table

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 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 4.3 由表4.3映射的类二部图

Fig 4.3 Bigraph-similar mapped by table 4.3

再得到的元组边为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4.4所示：

表4.4 多敏感属性L-多样性（L=3）发布结果

Table 4.4 Release results of L- diversity (L=3)

QIT（准标识表） ST（敏感属性表）

|  |  |
| --- | --- |
| 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> |

|  |  |  |
| --- | --- | --- |
| 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 |

## 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。该数据集也是大多数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4.5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4.5 实验数据集信息

Table 4.5 Experimental data set information

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

表4.6 实验中采用的复合敏感属性

Table 4.6 Complex sensitive attributes used in the experiment

|  |  |
| --- | --- |
| 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算法得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[30]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4.4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4.5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4.4与图4.5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4.4 算法隐匿率随着数据量的变化

Fig 4.4 The SuppRatio of the algorithm changes with the amount of data

图 4.5 算法附加信息损失度随着数据量的变化

Fig 4.5 The additional information loss to algorithm varies with the amount of data

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4.6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4.6 算法数据隐匿率随着多样性参数L的变化

Fig 4.6 The SuppRatio of the algorithm changes with the diversity parameter L.

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4.7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 4.7 BES算法执行时间随着敏感属性维数d的变化

Table 4.7 The execution time of BES algorithm changes with the dimension of sensitive attribute d.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 2维敏感属性 | | 3维敏感属性 | 4维敏感属性 | 5维敏感属性 |
| 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。

# 5 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。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各敏感属性值的不同敏感度取值和敏感属性自身的敏感度取值，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4.3为例，计算得到所有元组边的敏感度权值。表5.1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 敏感度权值表

Table 5.1 Sensitivity weight table

|  |  |  |  |
| --- | --- | --- | --- |
| 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4.3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5.2所示：

表5.2 带权元组边集合表

Table 5.2 Edge set table with weighted tuple

|  |  |  |  |
| --- | --- | --- | --- |
| 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[30]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） (5.1)

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|S(Disease)|=5。

最大敏感度阈值：

(5.2)

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. 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 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
7. 将ti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|]
    * + - 1. f ti加入到GSj中GSj依然满足多敏感属性（L，）-diversity个性化分组
15. 元组ti加入到GSj中，并将ti从RT中移除
16. 隐匿RT中所有剩余的元组
17. 将GS中所有分组以QIT，ST形式发布

图5.1 带权类二部图

Fig 5.1 Weighted Bigraph-similar

我们依然以表4.3为待发布数据表，根据表5.1中的敏感度权值设置来执行算法，算法处理第一步得到表5.2的带权元组边集合，映射成带权类二部图如5.1所示。得到带权元组边集合WTE。

令，则根据表5.1敏感度权值的设置和5.1.2节中式（5.2）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5.3所示：

表5.3 （3，）-diversity个性化敏感数据发布表

Table 5.3 Publishing table meets (3，）-diversity personalized

|  |  |
| --- | --- |
| 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> |

QIT(准标识属性表) ST(敏感属性表)

|  |  |  |
| --- | --- | --- |
| 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—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，）-diversity个性化数据发布表，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5.3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的条件，从而造成虽然得到的分组虽然满足（L，）-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，）-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={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依次递减，表示为：WTE1WTE2，WTE3，···，WTEL

关于带权元组边集合L等级划分如图5.2所示。



图 5.2 WTE权值等级L-划分

Fig 5.2 WTE Weight class L- Division

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{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5.4。

表5.4 降序带权元组边集合（WTE）表

Table 5.4 Descending weighted tuple edge set(WTE) table

|  |  |  |  |
| --- | --- | --- | --- |
| 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5.5的最终数据发布表。

根据最终发布表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}，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，按元组敏感度分段取元组，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5.5 L-WSES分组算法得到最终数据发布表

Table 5.5 Publication table through L-WSES

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 |

|  |  |
| --- | --- |
| 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> |

## 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5.6与5.7。

表5.6 不同敏感属性个数下敏感属性自身敏感度设定值

Table 5.6 The different sensitivity value ofdifferent sensitive attributes

|  |  |  |
| --- | --- | --- |
| 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5.7 各敏感属性值敏感度设定值

Table 5.7 Sensitive attribute value sensitivity setting values

（a）

|  |  |  |  |
| --- | --- | --- | --- |
| Occupation | Weight | Education | Weight |
| Tech-support | 0.6 | Bachelors | 0.3 |
| Craft-repair | 0.5 | Some-college | 0.2 |
| Sales | 0.5 | 11th | 0.7 |
| Exec-managerial | 0.5 | HS-grad | 0.1 |
| Prof-specialty | 0.5 | Prof-school | 0.7 |
| Handlers-cleaners | 0.6 | Assoc-acdm | 0.4 |
| Machine-op-inspct | 0.6 | Assoc-voc | 0.4 |
| Adm-clerical | 0.5 | 9th | 0.7 |
| Farming-fishing | 0.6 | 7th-8th | 0.7 |
| Transport-moving | 0.5 | 12th | 0.7 |
| Priv-house-serv | 0.7 | Masters | 0.5 |
| Protective-serv | 0.7 | 1st-4th | 0.7 |
|  |  | 10th | 0.7 |
| Armed-Forces | 0.8 | Doctorate | 0.7 |
|  |  | 5th-6th | 0.7 |
|  |  | Preschool | 0.8 |

（b）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Work-class | Weight | Marital-status | Weight | Race | Weight |
| Private | 0.2 | Married-civ-spouse | 0.3 | White | 0.5 |
| Self-emp-not-inc | 0.6 | Divorced | 0.7 | Asian-Pac-Islander | 0.5 |
| Self-emp-inc | 0.6 | Never-married | 0.5 | Amer-Indian-Eskim | 0.5 |
| Federal-gov | 0.7 | Separated | 0.7 | Black | 0.5 |
| Local-gov | 0.6 | Widowed | 0.8 | Other | 0.5 |
| State-gov | 0.6 | Married-spouse-absent | 0.8 |  |  |
| Without-pay | 0.8 | Married-AF-spouse | 0.8 |  |  |
| Never-worked | 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，由于可变参数的取值不同，直接影响（L，）- 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5.3给出了数据量n=5k，敏感属性维数d=3，多样性参数L=3时，不同取值情况下，数据隐匿率的变化情况。可以看到随着取值的逐渐增大，L-SWES分组算法得到的发布数据隐匿率呈下降趋势，这是因为取值与模型定义中分组敏感度阈值成正相关，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。当取值大于等于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5.3 可变参数β取值对数据发布隐匿率的影响（n=5k,L=3,d=3）

Fig 5.3 The SuppRatio of the algorithm changes with the β（n=5k,L=3,d=3）

通过图5.3实验结果，针对多敏感属性的（L，）- diversity个性化匿名模型，选取=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，）- 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5.5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，）- 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 数据隐匿率随着数据量的变化（L=3，d=3）

Fig 5.4 The SuppRatio changes with the amount of data(L=3，d=3)

图5.5 附加信息损失度随着数据量的变化（L=3，d=3）

Fig 5.5 The additional information loss varies with the amount of data（L=3,d=3）

图5.6 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5.6 数据隐匿率随着多样性参数L的变化（n=5k,d=3）

Fig 5.6 The SuppRatio changes with the diversity parameter L(n=5k,d=3)

表5.8 不同敏感属性维数下数据隐匿率的变化（n=5K，L=3）

Table 5.8 The SuppRatio changes with the different dimensions of Sensitive attribute

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 算法 | 2维敏感属性 | 3维敏感属性 | 4维敏感属性 | 5维敏感属性 |
| BES | 0.12% | 0.12% | 0.23% | 45% |
| L-SWES | 0.12% | 1.3% | 4.6% | 64% |

表5.8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，）- 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。

# 6 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，）-diversity个性化数据发布模型。该模型通过设置调节个性化参数从而得到合适的模型中分组最大阈值参数，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，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。

③ 对以上提出的（L，）-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，）-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，）-diversity个性化数据发布模型依赖个性化参数的设置，在特定条件下如何快速准确定义个性化的值，关于个性化参数的定义与取值，可专门做研究，从而更加完善（L，）-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。

# 致 谢

# 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. 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. 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. Fung B C M, Wang K, Chen R, et al. Privacy-preserving data publishing: A survey of recent developments[J]. Acm Computing Surveys, 2010, 42(4):14.
19. 徐勇, 王浩, 李东勤. 数据发布领域匿名隐私保护相关技术研究[J]. 情报杂志, 2011(8):128-133.
20. 李立. 面向相关多敏感属性的隐私保护方法[D]. 河北大学, 2011.
21. 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.
22. 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.
23. 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)．
24. 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．
25. Samarati P．Protecting respondents’identities in microdata release[C]．In Proc of the TKDE．2001：1 01 0-1027．
26. 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
27. Soria-Comas,Jordi.Domingo-Ferrer,Josep. Probabilistic k-anonymity through microaggregation and data swapping[C].2012 IEEE International Conference on Fuzzy Systems,2012:8.
28. G.T.Duncan and D.Lamber. Disclosure-limited data dissemination[J].Journal of the American Statistical Association,1986,81:10-28.
29. 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
30. 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.
31. 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
32. Xiao Xiaokui,Tao Yufei. Personalized Privacy Preservation[C] / / Proceedings of ACM SIGMOD International Conferenceon on Management of Data.New York,USA,2006:229-240.
33. 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.
34. 杨静, 王波. 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[J]. 计算机研究与发展, 2012, 49(12):2603-2610.
35. 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.
36. Qiyuan Gong，Ming Yang，Zhenguo Chen，Junzhou Luo．Utility Enhanced Anonymization for Incomplete Microdata
37. 徐龙琴, 刘双印. 语义相似和多维加权的联合敏感属性隐私保护[J]. 计算机应用, 2011, 31(4):999-1002
38. 孙艳正, 陈伟鹤, 詹永照. 基于分类敏感属性语义距离的隐私保护模型[J]. 计算机应用研究, 2011, 28(10):3823-3827.
39. 王波, 杨静. 数据发布中的个性化隐私匿名技术研究[J]. 计算机科学, 2012, 39(4):168-171.
40. Poulis G, Skiadopoulos S, Loukides G, et al. Distance-Based k^m-Anonymization of Trajectory Data[C]// IEEE, International Conference on Mobile Data Management. IEEE Computer Society, 2013:57-62.
41. Lefevre K, Dewitt D J, Ramakrishnan R. Mondrian Multidimensional K-Anonymity[C]// International Conference on Data Engineering. IEEE, 2006:25-25.
42. Xu J, Zhang Z, Xiao X, et al. Differentially private histogram publication[J]. Vldb Journal, 2013, 22(6):797-822.
43. Liang C, Fan J. a-diversity and k-anonymity Big Data Privacy Preservation Based on Micro-aggregation[J]. Netinfo Security, 2015.
44. 陶丽, 周萍. 一种新的基于敏感属性值划分的数据隐私保护算法[J]. 科技通报, 2016, 32(11):193-197.
45. Sweeney L．K—anonymity：A model for protecting privacy．International Journal on Uncertainty，Fuzziness，and Knowledge-Based Systems，2002，10(5)：557—570
46. Samarati P，Sweeney L．Generalizing data to provide anonymity when disclosing information
47. Zhang J, Cormode G, Procopiuc C M, et al. PrivBayes: private data release via bayesian networks[J]. 2014, 42(4):1423-1434.
48. 冯登国, 张敏, 李昊. 大数据安全与隐私保护[J]. 计算机学报, 2014, 37(1):246-258.
49. 王璐, 孟小峰. 位置大数据隐私保护研究综述[J]. 软件学报, 2014, 25(4):693-712.
50. 刘雅辉, 张铁赢, 靳小龙,等. 大数据时代的个人隐私保护[J]. 计算机研究与发展, 2015, 52(1):229-247.