IJRRECS/August 2013/Volume-1/Issue-4/141-146 **ISSN 2321-5461**

INTERNATIONAL JOURNAL OF REVIEWS ON RECENT ELECTRONICS AND COMPUTER SCIENCE

STRATEGY TO PUBLISH HIGH DIMENSIONAL MICRODATA ENSURING PRIVACY AND UTILITY

B.Uma[1], T.Swapna[2], Dr. K.J Sarma[3]

[1]M.Tech Student, Dept of CSE, TRR Engineering College, Patancheru, R.R Dist, A.P, India

[2]Associate Professor, Dept of CSE, TRR Engineering College, Patancheru, R.R Dist, A.P, India

[3]Professor & HOD, Dept of CSE, TRR Engineering College, Patancheru, R.R Dist, A.P, India

ABSTRACT:

In this paper we propose a strategy for ensuring privacy of high dimensional micro data. Sharing the micro data in the internet without violating the privacy policy is really challenging. The data normally contains personal details which are personally identifiable to anyone, thus violate the Privacy. Privacy protection is very important in data disclosure. Many organizations share micro data for research purposes, and necessary steps should be taken to make sure that an adversary cannot predict sensitive details regarding a particular individual with high confidence. Lots of effort is being undertaken recently in the area of data publishing to share data without compromising on the data privacy and data utility, especially for the high dimensional data. Anonymization approaches such as Generalization loses data utility while Bucketization, does not prevent membership disclosure. We propose a strategy called slicing to handle high dimensional micro data which gives better data utility and data privacy .This strategy breaks the association among the uncorrelated attributes to provide better privacy and preserves the connection between highly correlated attributes to provide better utility.

Keywords: slicing, privacy preservation, data utility, high dimensional data.

I. INTRODUCTION

In the recent years lots of research is dedicated to the publishing of high dimensional microdata ensuring data privacy and data utility [7]. Microdata is a set of tuples each of which contains details about an individual entity, such as a person, a household, or an organization. There are many microdata anonymization approaches that have been proposed. The most popular ones are generalization for k-anonymity and bucketization for l-diversity. In the above mentioned anonymization approaches, the attributes are partitioned into three categories [7]: 1. Some attributes are identifiers that can uniquely identify an individual, such as Name or Social Security Number. 2. Some attributes are Quasi Identifiers (QI), which the adversary may already know (possibly from other publicly available databases)

and which, when taken together, can potentially identify an individual, e.g., Birthdate, Sex, and Zipcode. 3. Some attributes are Sensitive Attributes (SAs), which are unknown to the adversary and are considered sensitive, such as Disease and Salary. In both generalization and bucketization, the identifiers are removed from the data and then partitioned into buckets. The two approaches differ in the next step[7]. Generalization modifies the QI-values in each bucket into "less specific but semantically consistent" values so that tuples in the same bucket cannot be distinguished by their QI values. In bucketization, one separates the SAs from the QIs by randomly permuting the SA values in each bucket. The anonymized data consist of a set of buckets with permuted sensitive attribute values.

2. RELATED WORKS

In this chapter we discuss about the literature survey and related works done in privacy preserving high dimensional microdata and their techniques. The major disadvantage of Generalization is: [7] it loses considerable amount of information, especially for high- dimensional data. And also, Bucketization does not prevent membership disclosure and does not apply for data that do not have a clear separation between quasi-identifying attributes and sensitive attributes. Generalization loses considerable amount of information, especially for high- dimensional data. C.Aggarwal [1] initially proposed k-anonymity and curse of dimensionality concept. Where the author [1] proposed privacy preserving anonymization technique where a record is released only if it indistinguishable from k other entities of data.

In the paper [1] the authors [1] show that when the data contains a large number of attributes which may be considered quasi-identifiers, it becomes difficult to anonymize the data without an unacceptably high amount of information loss. This is because an exponential number of combinations of dimensions can be used to make precise inference attacks, even when individual attributes are partially specified within a range. In the paper [1] they provide an analysis of the effect of dimensionality on k- anonymity methods. The authors of [1] conclude that when a data set contains a large number of attributes they are open to inference attacks, and also the author [1] faced with a choice of either completely suppressing most of the data or losing the desired level of anonymity. Thus, the work showed that the curse of high dimensionality also applies to the problem of privacy preserving data mining.

A. Blum[2] et.al., proposed a new framework for practical privacy and they named it as SULQ framework. The authors of [2] consider a statistical database in which a trusted administrator introduces noise to the query responses with the goal of maintaining privacy of individual database entries. Results of Dinur, Dwork, and Nissim show that a strong form of privacy can be maintained using a surprisingly small amount of noise – much less than the sampling error – provided the total number of queries is sublinear in the number of database rows. The assumption of sublinearity becomes reasonable as databases grow increasingly large. The authors [2] extend the work in two ways. First, they [2] modify the privacy analysis to real-valued functions and arbitrary row types, as a consequence greatly improving the bounds on noise required for privacy. Second, they [2] examine the

computational power of the SuLQ primitive.

They [2] show that it is very powerful indeed, in that slightly noisy versions of the following computations can be carried out with very few invocations of the primitive: principal component analysis, k means clustering, the Perceptron Algorithm, the ID3 algorithm, and all algorithms that operate in the in the statistical query learning model. J. Brickell [3] introduced a new anonymization technique called the cost of privacy. In this work, Re-identification is a major privacy threat to public datasets containing individual records. Many privacy protection algorithms rely on generalization and suppression of "quasi-identifier" attributes such as ZIP code and birthdate. Their objective is usually syntactic sanitization: for example, k-anonymity requires that each "quasi-identifier" tuple appear in at least k records, while l- diversity requires that the distribution of sensitive attributes for each quasi-identifier have high entropy. The utility of sanitized data is also measured syntactically, by the number of generalization steps applied or the number of records with the same quasi- identifier. In the paper [3], query generalization and suppression of quasi-identifiers offer many benefits over trivial sanitization which simply separates quasi- identifiers from sensitive attributes. Previous work showed that k-anonymous databases can be useful for data mining, but k-anonymization does not guarantee any privacy. By contrast, they measure the tradeoff between privacy and utility. The results demonstrate that even modest privacy gains require almost complete destruction of the data-mining utility. In most cases, trivial sanitization provides equivalent utility and better privacy than k-anonymity, l- diversity, and similar methods based on generalization and suppression. A multidimensional technique was proposed by B.C. Chen et. al [4], which they named as Skyline based technique. Privacy is an important issue in data publishing. I.Dinur [5] proposed another technique of revealing information while preserving privacy. The authors [5] examine the tradeoff between privacy and usability of statistical databases.

Let us consider microdata such as census data or medical data. Typically, microdata are stored in a table, and each row corresponds to an individual. Each record has a number of attributes[7], which can be divided into the following three categories: 1. Identifier: Identifiers are attributes that clearly identify individuals. Examples include Social Security Number and Name. 2. Quasi-Identifier: Quasi-identifiers are attributes whose values when taken together can potentially identify an individual. Examples include Zip- code, Birthdate, and Gender. An adversary may already know the QI values of some individuals in the data. This knowledge can be either from personal contact or from other publicly available databases (e.g., a voter registration list) that include both explicit identifiers and quasi- identifiers. 3. Sensitive Attribute: Sensitive attributes are attributes whose values should not be associated with an individual by the adversary. Examples include Disease and Salary.

TABLE1 – ORIGINAL TABLE

Name	Age	Gender	Zipcode	Disease
Anil	20	Male	500010	Flu
Bhuvana	25	Female	500020	Fever
Chaitanya	23	Male	500015	Flu
Geetha	44	Female	500020	Cancer
Mahesh	55	Male	500022	AIDS
Naveen	58	Male	500022	Fever
Sudha	62	Female	500020	Heart Attack
Vishwanat	58	Male	500018	Diabetes

TABLE 2: DATA AFTER GENERALIZATION

Age	Gender	Zipcode	Disease
[20 - 62]	Male	50*	Flu
[20 - 62]	Female	50*	Fever
[20 - 62]	Male	50*	Flu
[20 - 62]	Female	50*	Cancer
[20 - 62]	Male	50*	AIDS
[20 - 62]	Male	50*	Fever
[20 - 62]	Female	50*	Heart Attack
[20 - 62]	Male	50*	Diabetes

TABLE 3: DATA AFTER BUCKETIZATION

Age	Gender	Zipcode	Disease
[20 - 45]	*	50*	Flu
[20 - 45]	*	50*	Fever
[20 - 45]	*	50*	Flu
[20 - 45]	*	50*	Cancer
[46 -70]	*	50*	AIDS
[46 -70]	*	50*	Fever
[46 -70]	*	50*	Heart Attack
[46 -70]	*	50*	Diabetes

3. PROPOSEDWORK:

In this paper, an approach called slicing is used and its impact on the data utility, data privacy of publishing the high dimensional microdata is analyzed. Slicing has many advantages when compared to generalization and bucketization approaches. It preserves better data utility than generalization[7]. It preserves more attribute correlations with the Sensitive Attributes than bucketization. Slicing can be effectively used for preventing attribute disclosure, based on the privacy requirement of l-diversity. An efficient algorithm for computing the sliced table that satisfies l-diversity is analyzed which partitions attributes into columns, applies column generalization, and partitions tuples into buckets.

Attributes that are highly correlated are kept in the same column; this preserves the correlations between such attributes[7]. The associations between uncorrelated attributes are broken; this provides better privacy as the associations between such attributes are less frequent and potentially identifying. It is also analyzed how slicing algorithm prevents membership disclosure. A bucket of size k can potentially match k_c tuples where c is the number of columns. Because only k of the k_c tuples are actually in the original data, the existence of the other $k_c - k$ tuples hides the membership information of tuples in the original data [7].

Slicing partitions the dataset both vertically and horizontally. Vertical partitioning is done by grouping attributes into columns based on the correlations among the attributes. Each column contains a subset of attributes that are highly correlated. Horizontal partitioning is done by grouping tuples into buckets. Finally, within each bucket, values in each column are randomly permutated (or sorted) to break the linking between different columns. The basic idea of slicing is to break the association cross columns, but to preserve the association within each column. This reduces the dimensionality of the data and preserves better utility than generalization and bucketization[7]. Slicing preserves utility

because it groups highly correlated attributes together, and preserves the correlations between such attributes. Slicing protects privacy because it breaks the associations between uncorrelated attributes, which are infrequent and thus identifying. Note that when the dataset contains QIs and one SA, bucketization has to break their correlation; slicing, on the other hand, can group some QI attributes with the SA, preserving attribute correlations with the sensitive attribute.

Slicing Algorithm: An effective slicing algorithm to obtain ℓ- diverse slicing is used. For a given a micro data table T and two factors c and ℓ, the algorithm calculates the sliced table that has c columns and satisfies the privacy requisite of ℓ- diversity. Our algorithm has three steps: attribute partitioning, column generalization and tuple partitioning.

Attribute Partitioning: Our algorithm partitions attributes so that highly related attributes are in the same column. This is better for utility as well as privacy.

Column Generalization: Here, records are generalized to ensure minimum frequency requisite. Tuple Partitioning: In this step, the records are grouped into buckets. Membership Disclosure Protection: The adversary can inspect the data and check the QI values to find out the presence of a certain individual's data. So, it is necessary that, in the anonymized data set, a record in the real information should have same occurrence as a record which is not present in the original information. Otherwise, by investigating their occurrences in the data that is anonymized, the adversary can distinguish records in the real information from records that are not present in the original information which leads to membership disclosure.

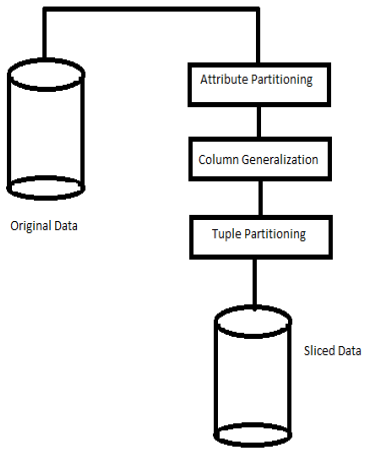

Fig 1: Flow diagram

Sliced Data : Slicing partitions attributes into columns and reduces the dimensionality of the data. Each column of the table can be viewed as a sub-table with a lower dimensionality. Slicing is also different from the approach of publishing multiple independent sub- tables in that these sub-tables are linked by the buckets in slicing.

TABLE 4 SLICED DATA

Age, Gender	Zip, Disease
20, Male	500010,Fever
25, Female	500020,Flu
23, Male	500015,Cancer
44,Female	500020,Flu
55, Male	500022,Fever
58, Male	500022, Heart Attack
62, Female	500020, Diabetes
58, Male	500018, AIDS

Table 4 shows the sliced table corresponding to the original table data, Table 1. It contains two columns, first one with attributes {Age, Gender} and the

second with attributes {Zipcode, Disease}. The data is split to two buckets and within each bucket the values in each column are randomly permuted to break the linking between different columns.

4. ALGORITHM

The Following steps are used to slice the given table.

1. Load the original data to be sliced.

2. Apply Attribute partition and column generalization procedures.

3. Send to Tuple partition and buckets functionality.

4. Apply Slicing approach.

5. Process Column generalization.

6. Process matching buckets.

7. Duplicate an attribute in more than one column.

8. Store the Sliced output data.

5. CONCLUSION & FUTURE WORK:

From the theories and implementation it is proved that Slicing overcomes the limitations of existing techniques of generalization and bucketization while preserving better utility and protecting against privacy threats. Slicing can also be used to prevent attribute disclosure and membership disclosure. It is observed that, before anonymizing the data, one can analyze the data characteristics and use the characteristics in data anonymization. As our future work we plan to design more effective tuple grouping algorithms. The trade-off between column generalization and tuple partitioning is the subject of future work. The design of tuple grouping algorithms is left to future work.

REFERENCES

[1] C. Aggarwal, "On k-Anonymity and the Curse of Dimensionality," Proc. Int'l Conf. Very Large Data Bases (VLDB), pp. 901-909, 2005.

[2] A. Blum, C. Dwork, F. McSherry, and K. Nissim, "Practical Privacy: The SULQ Framework," Proc. ACM Symp. Principles of Database Systems (PODS), pp. 128-138, 2005.

[3] J. Brickell and V. Shmatikov, "The Cost of Privacy: Destruction of Data-Mining Utility in Anonymized Data Publishing," Proc. ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), pp. 70-78, 2008.

[4] B.-C. Chen, K. LeFevre, and R. Ramakrishnan, "Privacy Skyline: Privacy with Multidimensional Adversarial Knowledge," Proc. Int'l Conf. Very Large Data Bases (VLDB), pp. 770-781, 2007.

[5] H. Cramt'er, Mathematical Methods of Statistics. Princeton Univ. Press, 1948.

[6] I. Dinur and K. Nissim, "Revealing Information while Preserving Privacy," Proc. ACM Symp. Principles of Database Systems (PODS), pp. 202-210, 2003.

[7]"Slicing: A New Approach for Privacy Preserving Data Publishing", Tiancheng Li, Ninghui Li, Jian Zhang and Ian Molloy, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 24, NO. 3, pp. 561-574, MARCH 2012.