ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

IJESRT
INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

A Novel Data Anonymization Technique for Privacy Preservation of Data Publishing

P.Sreevani*, Dr.P.Niranjan, Dr. P.Shireesha

* M.Tech Department of Computer Science, Kakatiya Institute of Technology and Science, Warangal, India

Professor, Department of Computer Science, Kakatiya Institute of Technology and Science, Warangal, India

Professor, Department of Computer Science and Engineering,Kakatiya Institute of Technology and Science Warangal, India

Abstracts

Most enterprises are actively collecting and storing data in large databases. Many of them have recognized the potential value of data as an information source for making business decisions. Privacy preservation is important one for publishing the information containing individual specific records. In generally information about individual's records will violate the privacy. So many techniques have been introduced for preserving the privacy. The existing systems have been designed with the anonymization techniques of generalization and bucketization. Those methods were revealed the privacy of individual data to the adversaries. Generalization involved considerable loss of the data and bucketization method does not protection from membership disclosure and there is no clear separation between sensitive attributes and quasi identifier attributes. The proposed system of Slicing with Tuple grouping algorithm. Partitioned The data both horizontally and vertically. It provides better information utility than generalization and protection from membership disclosure and also can handle in high dimensional data. The Government agencies, organization and companies are shared the data and publication of the data for research purposes. This paper focuses on effective method that can be used for providing better data utility and can handle high level dimensional data.

Keywords: Micro data, Sensitive information, Data anonymization, Data publishing, Data security, Privacy Preservation.

Introduction

The Data mining is the extracting the meaningful information from the large data sets such as Micro data, data warehouse contains records each of which contains information about an individual entity. like as a person or an organization or household. Several micro data anonymization techniques have been introduced for this purpose. The most popular generalization for k-anonymity and bucketization for l-diversity. In both approaches attributes are partitioned into three categories-

a) Some attributes are identifiers that can uniquely identify an individual like Name or Social Security Number;

b) Some attributes are Quasi Identifiers, which the adversary may already know and when taken together can potentially identify an individual.

c) Some attributes are Sensitive Attributes which are unknown to the adversary and are considered sensitive like Salary. Generally when the micro data publishing the various attacks occurred like record linkage model attack and attribute linkage model attack. So avoid these attacks several anonymization techniques was introduced. In both generalization and bucketization removes the identifiers from the data and also partitions tuples into buckets. Buckets contain the subset of tuples. Generalization transforms the QI values in each bucket into less specific but semantically consistent values. So that tuples of the same bucket cannot be distinguished by their QI values. In bucketization separates the SAs from the QIs but randomly permuting the SA values in each bucket. The major limitation of the traditional approach of k Anonymity is that link the external data with shared data. In generalization all the attributes are suppressed until each row is identical. It is used for prevent identifier disclosure but it is not guarantee to the entire privacy and lose the information in high dimensional data. In bucketization technique all the sensitive information denoted the values are well represented. This technique has several limitations first one is does not prevent membership disclosure.

Because bucketization publishes the quasi identifier values in their original forms, an adversary can find out whether an individual has a record in the already published data or not. The proposed Slicing algorithm with tuple grouping algorithm is partitioned the data both horizontally and vertically. The random values are permutated within each bucket and also can handle in high dimensional data .It is more data utility than generalization and bucketization.

Various anonymization techniques

Anonymization in the existing algorithms and various techniques would possibly suit the real time databases. We discuss about the existing algorithms for Privacy preserving of micro data publishing.

A. Generalization:

There are several types of methods for recordings for generalization data. The recoding that preserves the most information is local recoding method. In local recoding one first groups tuples into buckets and then for each bucket replaces all values of one attribute with a generalized value Such a recoding is local because the same attribute value may be generalized. Differently when they appear in different buckets. We now show that slicing preserves more information than such a local recoding approach assuming that the same tuple partition is used. We achieve this by showing that slicing is better than the following enhancement of the local recoding approach. Rather than using a generalized value to replace more specific attribute values one uses the multi set of exact values in each bucket. The multi set of exact values provides more information about the distribution of values in each attribute than the generalized interval. Therefore using multi sets of exact values preserves more information than generalization.

B. Bucketization:

The Bucketization is to partition the tuples in T into buckets and then to separate the sensitive attribute from the non sensitive ones by randomly permuting the sensitive attribute values within each bucket. The sanitized data then consists of the buckets with permuted sensitive values. In this paper we use bucketization as the method of constructing the published data from the original table T, although all our results hold for full-domain generalization as well. We now specify our notion of bucketization more formally. Partition the tuples into buckets and within each bucket we apply an independent random permutation to the column containing S-values. The resulting set of buckets denoted by B is then published. For example if the underlying table T then the publisher might publish bucketization for added privacy the publisher can completely mask the identifying attribute and may partially mask some of the other non-sensitive attributes. For a bucket B we use the following notation. While bucketization has better data utility than generalization it has several limitations. First bucketization does not prevent membership disclosure . Because bucketization publishes the QI values in their original forms an adversary can find out whether an individual has a record in the published data or not. As shown in 87 percent of the individuals in the United States can be uniquely identified using only three attributes. A micro data usually contains many other attributes besides those three attributes. This means that the membership information of most individuals can be inferred from the bucketized table. Second bucketization requires a clear separation between QIs and SAs. However in many data sets it is unclear which attributes are QIs and which are SAs. Third by separating the sensitive attribute from the QI attributes bucketization breaks the attribute correlations between the QIs and the SAs. Bucketization first partitions tuples in the table into buckets and then separates the quasi identifiers with the sensitive attribute by randomly permuting the sensitive attribute values in each bucket. The anonymized data consist of a set of buckets with permuted sensitive attribute values. In particular, bucketization has been used for anonymzing high-dimensional data. However, their approach assumes a clear separation between QIs and SAs. In addition because the exact values of all QIs are released, membership information is disclosed.

C. Slicing:

To improve the current state of the art in this paper, we introduce a novel data Anonymization technique called slicing. Slicing partitions the data set 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 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. 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 data set 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. The key intuition that slicing provides privacy protection is that the slicing process ensures that for any tuple there are generally multiple matching buckets. Slicing first partitions attributes into columns. Each column contains a subset of attributes. Slicing also partitions the tuples into buckets. Each bucket contains a subset of tuples. This horizontally partitions the table. Within each bucket, values in each column are randomly permutated to break the linking between different columns.

Related work

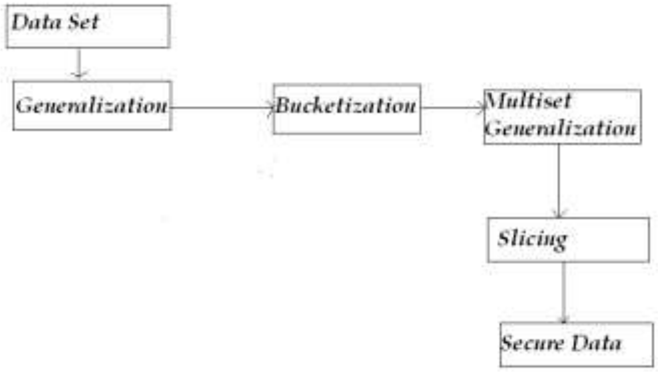
The Two popular Anonymization techniques are generalization and bucketization. Generalization, replaces a value with a "less-specific but semantically consistent" value. The main problems with generalization are:
1) It fails on high dimensional data due to the curse of dimensionality.
2) It causes too much information loss due to the uniform-distribution assumption.
Bucketization first partitions tuples in the table into buckets and then separates the quasi identifiers with the sensitive attribute by randomly permuting the sensitive attribute values in each bucket. The anonymized data consist of a set of buckets with permuted sensitive attribute values. In particular, bucketization has been used for anonymizing high-dimensional data. However, their approach assumes a clear separation between QIs and SAs. In addition, because the exact values of all QIs are released, membership information is disclosed. The key idea of slicing is to preserve correlations between highly correlated attributes and to break correlations between uncorrelated attributes thus achieving both better utility and better privacy. Third, existing data analysis methods can be easily used on the sliced data.

Problem definition and architecture

A. Problem definition:

Generally in privacy preservation there is a loss of security. The privacy protection is impossible due to the presence of the adversary's background knowledge in real life application. Data in its original form contains sensitive information about individuals. These data when published violate the privacy. The current practice in data publishing relies mainly on policies and guidelines as to what types of data can be published and on agreements on the use of
Published data. The approach alone may lead to excessive data distortion or insufficient protection. Privacy-preserving data publishing provides methods and tools for publishing useful information while preserving data privacy. Many algorithms like bucketization, generalization have tried to preserve privacy however they exhibit attribute disclosure. So to overcome this problem an algorithm called slicing is used.

B. Functional and slicing architecture:



C. Functional procedure

Step 1: Extract the data set from the database.
Step 2: Anonymity process divides the records into two.
Step 3: Interchange the sensitive values.
Step 4: Multistep values generated and displayed.
Step 5: Attributes are combined and secure data Displayed.

D. Slicing algorithm:

We then formalize slicing, compare it with generalization and bucketization, and discuss privacy threats that slicing can address. Generally in privacy preservation there is a loss of security. The privacy protection is impossible due to the presence of the adversary's background knowledge in real life application. Data in its original form contains sensitive information about individuals. These data when published violate the privacy. The current practice in data publishing relies mainly on policies and guidelines as to what types of data can be published and on agreements on the use of published data. The approach alone may lead to excessive data distortion or insufficient protection. Privacy-preserving data publishing provides methods and tools for publishing useful information while preserving data privacy. Many algorithms like bucketization, generalization have tried to preserve privacy however they exhibit attribute disclosure. So to overcome this problem an algorithm called slicing is used. This algorithm consists of three phases: attribute partitioning, column generalization, and tuple partitioning.

i. Attribute partitioning:

This algorithm partitions attributes so that highly correlated attributes are in the same column. This is good for both utility and privacy. In terms of data utility, grouping highly correlated attributes preserves the correlations among those attributes. In terms of privacy, the association of uncorrelated attributes presents higher identification risks than the association of highly correlated attributes because the associations of uncorrelated attribute values is much less frequent and thus more identifiable.

ii. Column generalization:

First column generalization may be required for identity/membership disclosure protection. If a column value is unique in a column, a tuple with this unique column value can only have one matching bucket. This is not good for privacy protection, as in the case of generalization bucketization where each tuple can belong to only one equivalence-class/bucket.

iii. Tuple partitioning:

The algorithm maintains two data structures:

1) A queue of buckets Q

2) A set of sliced buckets SB. Initially, Q contains only one bucket which includes all tuples and SB is empty. For each iteration, the algorithm removes a bucket from Q and splits the bucket into two buckets. If the sliced table after the split satisfies l-diversity, then the algorithm puts the two buckets at the end of the queue Q Otherwise, we cannot split the bucket anymore and the algorithm puts the bucket into SB. When Q becomes empty, we have computed the sliced table. The set of sliced buckets is SB.

The main part of the tuple-partition algorithm is to check whether a sliced table satisfies diversity gives a description of the diversity-check algorithm. For each tuple t, the algorithm maintains a list of statistics L (t) about t s matching buckets. each element in the list L(t) contains statistics about one matching bucket b, the matching probability p (t, B) and the distribution of candidate sensitive values d(t, B). The algorithm first takes one scan of each bucket b to record the frequency f(v) of each column value v in bucket b

$$P(t, s) = \sum_{e \in L(t)} e.p(t, B) * e.D(t, B)[s] \qquad (1)$$

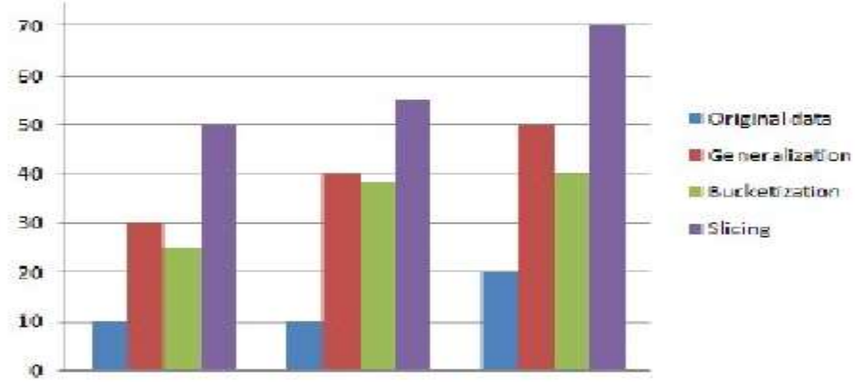
Then, the algorithm takes one scan of each tuple t in the table t to find out all tuples that match b and record their matching probability p(t, B) and the distribution of candidate sensitive values d(t, B) which are added to the list l(t). We have obtained, for each tuple t, the list of statistics L (t) about its matching buckets. A final scan of the tuples in t will compute the p (t, b) values based on the law of total probability.

Experimental evaluation

To allow direct comparison, we use the l-diversity for two anonymization techniques: slicing and optimized slicing for tuple grouping. This experiment demonstrates that:

1).slicing preserves better data utility than generalization;

2).slicing is more effective than bucketization in workloads involving the sensitive attribute; and 3) the sliced table can be computed efficiently. Both bucketization and slicing perform much better than generalization. We compare slicing with optimized slicing in terms of computational efficiency. We fix l= 5 and vary the cardinality of the data and the dimensionality of the data.



Conclusion

The implementation of previously existing systems provided clear view of the problem to be addressed. Slicing overcomes the limitations of generalization and bucketization and preserves better utility while protecting against privacy threats. Our experiments show that slicing preserves better data utility than generalization and is more effective than bucketization in workloads involving the sensitive attribute. First, in this paper, we consider slicing where each attribute is in exactly one column. An extension is the notion of overlapping slicing, which duplicates an attribute in more than one column. Our experiments show that random grouping is not very effective. The Proposed grouping algorithm is optimized L-diversity slicing check algorithm obtains the more effective tuple grouping and Provides secure data. Another direction is to design data mining tasks using the anonymized data computed by various anonymization techniques.

References

1. Aggarwal.C, “On K-Anonymity and the Curse of Dimensionality,” Proc. Int‟l Conf.Very Large Data Bases (VLDB), 2005.
2. Brickell.J and 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), 2008.
3. Ghinita.G,Tao.Y, and Kalnis.P, “OnThe Anonymization of Sparse High Dimensional Data,” Proc. IEEE 24th Int‟l Conf. Data Eng. (ICDE), 2008.
4. He.Y and Naughton.J, “Anonymization of Set-Valued Data via Top-Down, local Generalization,” Proc.IEEE 25th Int‟l Conf.Data Engineering (ICDE), 2009.
5. Inan.A, Kantarcioglu.M, and Bertino.e, “Using Anonymized Data for Classification,” Proc. IEEE 25th Int‟l Conf. Data Eng. (ICDE), pp. 429-440, 2009.
6. Li.T and Li.N, “On the Tradeoff between Privacy and Utility in Data Publishing,” Proc.ACM SIGKDD Int‟l Conf.Knowledge Discovery and Data Mining (KDD), 2009.
7. Li.N, Li.T, “Slicing: The new Approach for Privacy Preserving Data publishing”, IEEE Transaction on knowledge and data Engineering, vol.24, No, 3, March 2012.
8. Li.N,Li.T,and Venkatasubramanian.S,“t-Closeness: Privacy Beyond K-Anonymity And L-Diversity,”Proc.IEEE 23rd Intel Conf.Data Eng.(IDCE),2007.
9. Machanavajjhala.A, Gehrke.J, Kifer.D, and M.Venkitasubramaniam, “L-diversity privacy Beyond K- Anonymity”, Proc.IEEE 23 rd. Intel Conf.Data Eng,(ICDE),2007.
10. A. Inan, M. Kantarcioglu, and E. Bertino, “Using Anonymized Data for Classification,” Proc. IEEE 25th Int’l Conf. Data Eng. (ICDE)