A Rule based Slicing Approach to Achieve Data Publishing and Privacy

P. Bhuvaneswari[1] C. Saravanabhavan[2]
[1]PG Scholar [M.E.] [2]Assistant Professor
[1, 2]Department of Computer Science
[1, 2] AMS Engineering College, Namakkal

Abstract—several anonymization techniques, such as generalization and bucketization, have been designed for privacy preserving micro data publishing. Recent work has shown that generalization loses considerable amount of information, especially for high dimensional data. Bucketization, on the other hand, 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. The existing system proposed slicing concept to overcome the tuple based partition this has been done to overcome the previous generalization and bucketization. In this paper, present a novel technique called rule based slicing, which partitions the data both horizontally and vertically. We show that slicing preserves better data utility than generalization and can be used for membership disclosure protection. Another important advantage of slicing is that it can handle high-dimensional data. We show how slicing can be used for attribute disclosure protection and develop an efficient algorithm for computing the sliced data that obey the l-diversity requirement. The workload experiments confirm that slicing preserves better utility than generalization and is more effective than bucketization in workloads involving the sensitive attribute. The experiments also demonstrate that slicing can be used to prevent membership disclosure

Index Terms— Privacy preservation, data anonymization, data publishing, data security

I. INTRODUCTION

Individual entity such as a person a household or an organization several microdata anonymization techniques have been proposed. The most popular ones are generalization for k-anonymity and bucketization for L-diversity.

In both approaches, attributes are partitioned into three categories:

1) Some attributes are identifies that can uniquely identify an individual's such as Name or Social Security Number;
2) Some attributes are Quasi identifiers (Q1),, which the adversary may (possibly from other publicly available databases)and which, when taken together, can potentially identify an individual, e.g., Birthdate. Sex, and Zip code; - -
3) Some attributes are sensitive attributes (SAs), which are unknown to the adversary and considered sensitive, such as Disease and Salary.

In both generalization and bucketization, one first removes the identifiers from the data and then partitions tuples into buckets. The two techniques differ in the next step. Generalization transforms the Q1-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 Qls by randomly permuting the SA values in each bucket. The anonymized data consist of a set of buckets with permuted sensitive attribute values. PRIVACY-PRESERVING Publishing of micro-data has records each of which contains information about

II. MOTIVATION OF SLICING

In this paper, we introduce a novel data anonymization technique called slicing to improve the current state of the art. 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 (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. 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. Given a tuple $t = (v_i, v_2\ cr)$, where c is the number of columns and vi is the value for the ith column, a bucket is a matching bucket for t if and only if for each i $(1 < i < c)$, v_i appears at least once in the i'th column of the bucket. Any bucket that contains the original tuple is a matching bucket. At the same time, a matching bucket can be due to containing other tuples each of which contains some but not all vi's.

III. RELATED WORK

Two popular anonymization techniques are generalization and bucketization. Generalization [28], [30], replaces a value with a "less-specific but semantically

consistent" value. Three types of encoding schemes have been proposed for generalization: global recoding, regional recoding, and local recoding. Global recoding has the property that multiple occurrences of the same value are always replaced by the same generalized value. Regional record is also called multidimensional recoding (the Mondrian algorithm) which partitions the domain space into noninterest regions and data points in the same region are represented by the region they are in. Local recoding does not have the above constraints and allows different occurrences of the same value to be generalized differently. The main problems with generalization are: 1) it fails on high-dimensional data due to the curse of dimensionality and 2) it causes too much information loss due to the uniform-distribution assumption..

Slicing has some connections to marginal publication both of them release correlations among a subset of attributes. Slicing is quite different from marginal publication in a number of aspects. First, marginal publication can be viewed as a special case of slicing which does not have horizontal partitioning. Therefore, correlations among attributes in different columns are lost in marginal publication. By horizontal partitioning, Attribute correlations between different columns are preserved. Marginal publication is similar to overlapping vertical partitioning, which is left as our future work. Second, 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 (e.g., query answering) methods can be easily used on the sliced data.

Recently, several approaches have been proposed to anonymize transactional databases. Terrovitis et al. [31] proposed the k^m-anonymity model which requires that, for any set of m or less items, the published database contains at least k transactions containing this set of items. This model aims at protecting the database against an adversary who has knowledge of at most m items in a specific transaction. There are several problems with the anonymity model: 1) it cannot prevent an adversary from learning additional items because all k records may have some other items in common; 2) the adversary may know the absence of an item and can potentially identify a particular transaction; and 3) it is difficult to set an appropriate in value. He and Naughton [13] used k- anonymity as the privacy model and developed a local recoding method for anonymizing transactional databases. The k-anonymity model also suffers from the first two problems above. Xu et al. proposed an approach that combines k-anonymity and C-diversity but their approach considers a clear separation of the quasi identifiers and the sensitive attribute. On the contrary, slicing can be applied without such a separation.

Existing privacy measures for membership disclosure protection include differential privacy and & presence. Differential privacy has recently received much attention in data privacy. Most results on differential privacy are about answering statistical queries, rather than publishing microdata. A survey on these results can be found in. On the other hand, &presence assumes that the published database is a sample of a large public database and the adversary has knowledge of this large database. The calculation of disclosure risk depends on the choice of this large database. Finally, on attribute disclosure protection, a number of privacy models have been proposed, including e-diversity (α,k)-anonymity , and I-closeness . A few others consider the adversary's background knowledge.

IV. METHODOLOGIES

Our algorithm consists of three phases: Attribute partitioning, column generalization and tuple partitioning.
Attribute Partitioning:.

Pearson correlation coefficient and Mean-square contingency coefficient, we choose to use the mean-square contingency coefficient because most of our attributes are categorical. Two attribute A1, A2 with domain {v11, v12…v1d1} and {v21,v22…v2d2}respectively. Coefficient between A1 and A2 is defined as $Ø^2(A!,A2)=\{1/[\min(d1,d2)-1]\} \sum^{d1}\sum^{d2}\{(f_{ij}-f_{i.}f_{.j})^2/f_{i.}f_{.j}\}$.In this phase we first compute the correlations between pairs of attributes and then cluster attributes based on their correlations.

A. *Attribute clustering*

The distance between two attribute in the clustering space is defined as $d(A1,A2)=1- Ø^2(A1,A2)$which is in between of 0 and 1. We choose k=medoid method for the following reason. First, k-medoid method is very robust. Next one is data points are affect the cluster computed from the k-medoid method.

B. *Special Attribute Partitioning*

We adapt the above algorithm to partition attributes into c columns such that the sensitive column C, contains a attributes. We first calculate correlations between the sensitive attribute S and each QI attribute. Then, we rank the QI attributes by the decreasing order of their correlations with $ and select the top a - 1 QI attributes. Now, the sensitive column C, Consists of S and the selected QI attributes. All other QI attributes form the other c - 1 column using the attribute clustering algorithm.

C. *Column Generalization*

Column generalization is not a required phase; it can be useful in several aspects. First, column generalization may be required for identity/membership disclosure protection. If a column value is unique in a column (i.e., the column value appears only once in the column), a tuple with this unique column value can only have one matching bucket. This is not good for privacy protection
Second, when column generalization is applied, to achieve the same level of privacy against attribute disclosure, bucket sizes can be smaller .While column generalization may result in information loss, smaller bucket size allow better data utility.

D. Tuple Partitioning

In the tuple partitioning phase, tuples are partitioned into buckets

Algorithm tuple-partition (T,L)

Q= {T};SB= Ø
while Q is not empty
remove the first bucket B from Q;Q-{B}.
Split B into two buckets B1 and B2 as in Mondrain.
if(diversity-check(T,QU{B1,B2}USB,L)
Q=Q U {B1, B2}.
Else SB=SB U {B}.
Return (SB).

The algorithm maintain two data structures
1) a queue of buckets Q and
2) a set of sliced buckets SB.

Initially Q contains only one bucket which includes all tuples and SB is empty, In each iteration the algorithm removes 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. In this second phase, tuples are generalized to satisfy some minimal frequency requirement. We want to point out that column generalization is not an indispensable phase in our algorithm.

V. PROPOSED WORK

Introduce a novel data anonymization technique called slicing to improve the current state of the art. 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.

First formalize the idea of global background knowledge and propose the base model t-closeness which requires that the distribution of a sensitive in any equivalence class to be close to the distribution of the attribute in the overall table. Slicing protects privacy because it breaks the associations between uncorrelated attributes, which are infrequent and thus identifying. Slicing can be effectively used for preventing attribute disclosure.

Membership Disclosure protection

We propose two quantitative measures for the degree of membership protection offered by slicing. The first is the fake original ratio (FOR), which is defined as the number of fake tuples divided by the number of original tuples. The second measure is to consider the number of matching buckets for original tuples and that for fake tuples.

Attribute Disclosure protection

In our experiments we choose one attribute as the target attribute and all other attributes serve as the predictor attributes. We consider the performance of the anonymization algorithms in both learning the sensitive attribute occupation and learning a QI attribute education.

VI. CONCLUSION

The application works and creating databases for every hospital. The application works well and very helpful to get patients health record from anywhere. It is concluded that the application works well and satisfy the users. The application is tested very well and errors are properly debugged. The site is simultaneously accessed from more than one system. Simultaneous login from more than one place is tested. The site works according to the restrictions provided in their respective browsers. Further enhancements can be made to the application, so that the web site functions very attractive and useful manner than the present one. The speed of the transactions become more enough now.

VII. REFERENCES

[1] C. Aggarwal, "On k-Anonymity and the Curse of Dimensionality," Proc. flirt Conf. Very Large Databases (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 Coq 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] C. Dwork, "Differential Privacy," Proc. Jut'! Colloquium Automata, Languages and Programming (ICALP), pp. 1-12, 2006.
[8] C. Dwork, "Differential Privacy: A Survey of Results," Proc. Fifth Int'l Conf. Theory and Applications of Models of Computation (TAMC), PP· 1-19, 2008.
[10] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating Noise to Sensitivity in Private Data Analysis," Proc. Theory of Cryptography Conf. (TCC), pp. 265-284, 2006.
[11] J.H. Friedman, J.L. Bentley, and R.A. Finkel, "An Algorithm for Finding Best Matches in Logarithmic Expected Time," ACM Trans. Math. Software, vol. 3, no. 3, pp. 209-226, 1977.
[12] B.C.M. Fung, K. Wang, and P.S. Yu, "Top-Down Specialization for Information and Privacy Preservation," Proc. Intl Conf. Data Eng. (ICDE), pp. 205-216, 2005.
[13] G. Ghinita, Y. Tao, and P. Kalnis, "On the Anonyrnization of Sparse High-Dimensional Data," Proc. IEEE 24th Int'l Conf. Data Eng. (ICDE), pp. 715-724,2008.

