

Privacy Preservation of Sensitive Information using Slicing

Snehal Kalange, Research Scholar
Department of Computer Engineering,
Vidya Pratishthan College of Engineering, Baramati,413
133,University of Pune, Maharashtra, India.

D. M. Padulkar,Assistant Professor,
Department of Computer Engineering,
Vidya Pratishthan College of Engineering, Baramati,413
133, ,University of Pune, Maharashtra, India.

Abstract— Several approaches have been designed for privacy preserving data publishing. Slicing is one of them. In many organizations, there is need to publish or share their data. But, while publishing the data there is a requirement that privacy of sensitive information about the individual should not be violated. So, to preserve the privacy of sensitive information slicing with diversity approach is developed. Slicing partitions the dataset horizontally and vertically. Vertical partitioning is done by grouping attributes together into columns. Horizontal partitioning is done by grouping tuples into buckets. Slicing algorithm works in three phases which are attribute partitioning, column generalization and tuple partitioning. Attribute partitioning partitions attributes such that highly correlated attributes are in the same column. In Column generalization, tuples are generalized to satisfy minimal frequency requirement. In Tuple partitioning, tuples are partitioned into buckets. Also Slicing ensures attribute disclosure protection and membership disclosure protection and is able to handle high-dimensional data.

Keywords— *Slicing; Attribute Partitioning; Tuple Partitioning; Attribute Disclosure; Membership Disclosure.*

I. INTRODUCTION

Now days, collection of information by various organizations, government is used for knowledge-based decision making and analysis purpose. So, there is need to share and publish the collection of information. But, the data in its original form contains some sensitive information and people or organization does not want their sensitive information to be disclosed. Publishing data in its original form thus violates the individual privacy. So, to prevent this violation of privacy there should be some technique to publish the data in such a way that privacy is preserved and at the same time data analysis can be done effectively.

Slicing is one of the techniques for privacy preserving data publishing. In slicing, data attributes are partitioned into three categories which are: identifiers, Quasi Identifiers (QI), Sensitive Attributes (SAs). Identifiers uniquely identify an individual, such as Name or Social Security Number. Quasi Identifiers are the attributes 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. Sensitive Attributes are the attributes which are unknown to the adversary and are considered sensitive, such as Disease and Salary[3]. Slicing involves horizontal and vertical partitioning. Horizontal partitioning is done by grouping highly correlated attributes into columns and vertical partitioning is done by grouping tuples into buckets.

II. LITERATURE SURVEY

Earlier privacy preserving practices primarily relies on policies and guidelines to restrict the types of publishable data and on agreements on the use and storage of sensitive data[2]. The limitation of this approach is that it either distorts data excessively or requires a trust level that is impractically high in many data-sharing scenarios. After that many techniques have been developed, each one of them have some advantages and disadvantages. Some of the techniques are discussed in short below,

Generalization[3] is one of the anonymization techniques. Generalization transforms 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. That means the attribute values in the dataset are generalized. As the attribute values are generalized separately, therefore correlations between the attributes are lost. To generalize the values, it is required that the values in same bucket should be close to each other. But sometimes, in some datasets, attribute values are not close to each other. In such cases, generalization does not work. Also generalization does not work for high dimensional data as it losses considerable amount of information for high-dimensional data. In slicing, data is partitioned horizontally and vertically. That is, closely related attributes are grouped in one column and records in the dataset are partitioned into buckets. This partitioning reduces the dimensionality of the dataset. As slicing reduces dimensionality of the dataset, it can handle high-dimensional data. In this way, slicing can overcome the disadvantage of generalization by handling high-dimensional data.

In bucketization [4], one separates the Sensitive Attributes from the Quasi-identifying attributes by randomly permuting the Sensitive Attribute values in each bucket. The anonymized data consists of a set of buckets with permuted sensitive attribute values. Bucketization publishes the QI values in their original forms. As it publishes quasi-identifying attribute values in their original form, an adversary can find out whether an individual has a record in the published data or not. That means even if bucketization is performed on the dataset, re-identification of a record in the dataset may be possible. That is, though the dataset is bucketized, it may be possible for an adversary to re-identify an individual. Therefore, it is been clear that bucketization is unable to protect the dataset against the privacy threat membership disclosure. Slicing is a technique that can prevent the dataset against membership disclosure.

k-anonymity[5] is also one of the technique to preserve the privacy of microdata. k-anonymity demands that every tuple in the microdata table released be indistinguishably related to no fewer than k respondents. The basic requirement of k-anonymity principle is that, every record in the microdata which satisfies k-anonymity principle should match at least k other records. A table is said to be k-anonymous if it satisfies this k-anonymity principle. In k-anonymous table, every combination of quasi-identifying values there are at least k records that share these values. Though it makes difficult to re-identify an individual in the dataset, k-anonymity cannot protect the microdata from the homogeneity attack. K-anonymity suffers from homogeneity attack because it can create groups that can leak information due to the lack of diversity in sensitive attribute values. K-anonymity does not guarantee diversity among the sensitive attribute values. Slicing with ℓ -diversity principle overcome this disadvantage of k-anonymity by making sure that in each bucket of the table, there should be diversity among the sensitive attribute values.

III. BASIC TERMINOLOGY

A. Slicing

Slicing involves two major operations vertical partitioning and horizontal partitioning. Vertical partitioning is also known as attribute partitioning which partitions the dataset vertically by placing highly correlated attributes in one column. Horizontal partitioning is also known as tuple partitioning which partitions the dataset horizontally by grouping tuples in buckets. Also slicing breaks the association across the column and preserves the correlation within the column. Breaking association across the column preserves the privacy as the associations between the uncorrelated are infrequent and identifying. And preserving correlation within the column preserves the utility.

B. ℓ -Diversity

ℓ -diversity is a principle which ensures the diversity among the sensitive attributes in each bucket contained in the dataset. The ℓ -diversity principle is given below, A tuple t satisfies ℓ -diversity if for any sensitive value s ,

$$p(t, s) \leq 1/\ell \quad (1)$$

And a bucket is said to be ℓ -diverse if it contains at least ℓ well represented values for the sensitive attribute and if all the buckets in the table are ℓ -diverse, then such table is said to satisfy ℓ -diversity requirement. In this way, the ℓ -diversity principle ensures that if the table satisfies ℓ -diversity requirement then an adversary cannot predict the value of sensitive attribute with a probability greater than $1/\ell$. As described above, ℓ -diversity principle ensures ℓ well represented values for sensitive attribute in each bucket of a table. This makes it difficult for an adversary to predict the values of sensitive attribute which ultimately enhances the privacy of sensitive information. If ℓ -diversity principle is not applied, then there may be the case that all tuples in a bucket have same value for sensitive attribute. In such a case, adversary can accurately identify the value of sensitive attribute which violates the privacy requirement. Therefore, ℓ -diversity requirement is important to preserve the privacy of sensitive information.

IV. SYSTEM OVERVIEW

Slicing with ℓ -diversity is one of the privacy preserving data publishing technique. This approach preserves the privacy of sensitive information by protecting attribute disclosure and membership disclosure. The proposed system works as follows,

A. Attribute Disclosure Protection using ℓ -diversity

Attribute disclosure occurs when the released data makes it possible to infer the attributes of an individual more accurately. This disclosure should be protected. ℓ -diversity is adopted to achieve attribute disclosure protection. ℓ -diversity A bucket is ℓ -diverse if contains at least ℓ well represented values for the sensitive attribute S . A table is ℓ -diverse if every bucket is ℓ -diverse. A tuple t satisfies ℓ -diversity iff for any sensitive value s ,

$$p(t, s) \leq 1/\ell \quad (2)$$

where, $p(t, s)$ is called as the law of total probability which is the the probability that t takes sensitive value s which is calculated as,

$$p(t, s) = \sum_B p(t, B)p(s|t, B) \quad (3)$$

where, $p(t, B)$ is the probability that t is in bucket B which is calculated as,

$$p(t, B) = \frac{f(t, B)}{f(t)} \quad (4)$$

And $p(s|t, B)$ consists of candidate sensitive values. In this way, ℓ -diversity is used for attribute disclosure protection.

B. Slicing Algorithm

Slicing algorithm works in three phases, Attribute Partitioning, Tuple Partitioning.

- 1) *Attribute Partitioning*: Attribute Partitioning is done by grouping highly correlated attributes into a column. Mean-square Contingency Coefficient is used for correlation measurement, which is as follows,

$$\phi^2(A_1, A_2) = \frac{1}{\min(d_1, d_2) - 1} \sum_{i=1}^{d_1} \sum_{j=1}^{d_2} \frac{(f_{ij} - f_i f_j)^2}{f_i f_j} \quad (5)$$

where, A_1 and A_2 are the attributes among which the correlation is to be found out, d_1 and d_2 are the Domain sizes of A_1 and A_2 respectively. f_i is Fraction of occurrences of v_{1i} , f_j = Fraction of occurrences of v_{2j} and f_{ij} is Fraction of co-occurrences of v_{1i} and v_{2j}

After computing the correlations for each pair of attributes, clustering is used to partition attributes into columns. The distance between two attributes in the clustering space is defined as,

$$D(A_1, A_2) = \quad (6)$$

Two attributes that are strongly-correlated will have a smaller distance between the corresponding data points in our clustering space. And these strongly correlated attributes are placed in one column. In this way, attribute partitioning is done.

- 2) *Tuple Partitioning*: In tuple partitioning, tuples are partitioned into buckets. Tuple partitioning algorithm is as follows,

Algorithm tuple-partition

1. $Q = T$; $SB = \phi$
2. while Q is not empty
3. remove the first bucket B from Q ; $Q = Q - B$.
4. split B into two buckets B_1 and B_2
5. if diversity-check(T , $Q \cup B_1, B_2 \cup SB$, ℓ)
6. $Q = Q \cup B_1, B_2$.
7. else $SB = SB \cup B$.
8. return SB .

The algorithm maintains two data structures: a queue of buckets Q and 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 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 algorithm for ℓ -diversity check is as follows,

Algorithm ℓ -diversity check

1. for each tuple $t \in T$, $L[t] = \phi$
2. for each bucket B in T^*
3. record $f(v)$ for each column value v in bucket B .
4. for each tuple $t \in T$
5. calculate $p(t, B)$ and find $D(t, B)$.
6. $L[t] = L[t] \cup \{p(t, B), D(t, B)\}$.
7. for each tuple $t \in T$
8. calculate $p(t, s)$ for each s based on $L[t]$.
9. if $p(t, s) \geq 1/\ell$, return false.
10. return true.

The above algorithm maintains a list of statistics $L[t]$. Each element in the list $L[t]$ contains statistics about one matching bucket B . This algorithm ensures ℓ -diversity of every tuple in the dataset. In this way, sliced table satisfying ℓ -diversity requirement is produced using slicing algorithm.

Membership Disclosure Protection: Membership disclosure occurs when adversaries learn whether ones record is included in the published dataset. Slicing protects against membership disclosure as follows, Let D be the set of tuples in the original data and let D' be the set of tuples that are not in the original data. Let D^s be the sliced data. Given D^s and a tuple t , the goal of membership disclosure is to determine whether $t \in D$ or $t \in D'$. Slicing is an effective technique for membership disclosure protection. The tuples included in D' are called as the fake tuples. This inclusion of fake tuples makes it difficult to predict the membership of an individual in the dataset. In this way, slicing protects membership disclosure.

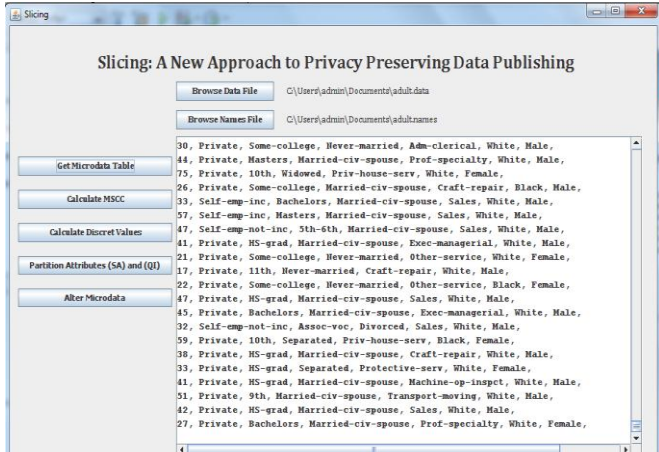
V. DATASET

We have used Adult Dataset, collected from UC Irvine machine learning repository which consists of data collected from the US census. The dataset initially consists of some tuples with missing attribute values. Such tuples are removed from the dataset in preprocessing stage. This OCC – 7 dataset consists of seven attributes which are age, work-class, education, marital-status, occupation, race, sex. Out of which age is continuous attribute and all the other attributes are categorical attributes. These attributes are then classified as Quasi-Identifying Attributes and Sensitive Attribute. Quasi-Identifying Attributes = {age, work-class, education, marital-status, race, sex} and Sensitive Attribute= {Occupation}.

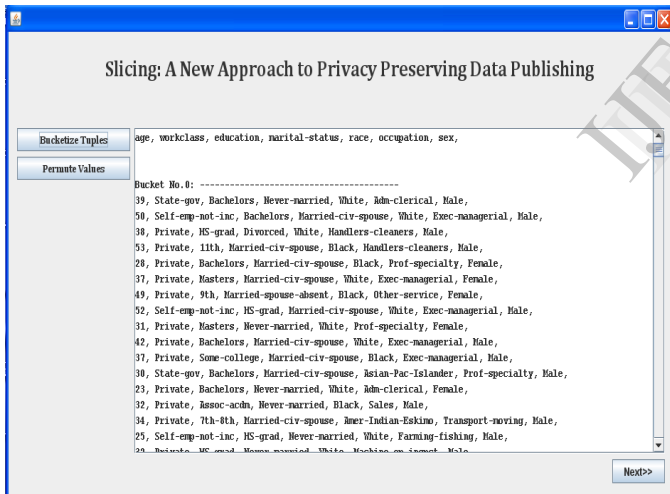
VI. RESULTS

To check the effectiveness of the technique 'Slicing with ℓ -diversity', we conducted experiments on the dataset mentioned above, that is Adult dataset. First the dataset is pre-processed and then Slicing is performed on this processed Adult dataset. Here, occupation is considered as the sensitive attribute. We preserved the privacy of this attribute using slicing with ℓ -diversity approach. ℓ -diversity ensured the diversity among the sensitive attribute values that is values of attribute, occupation. This diversity made it impossible for an adversary to predict the value of a sensitive attribute. We

partitioned the attributes in the dataset according to the distances between them. In Adult dataset, age is a continuous attribute so, kept in a separate column. Attributes work-class, marital-status and education are placed in one column as they are closer to each other. Then occupation and sex attributes have minimum distance between them, so they are placed in one column. At last, attribute race is at the maximum distance from all the other attributes therefore, it is placed in separate column. The results we got are given below,

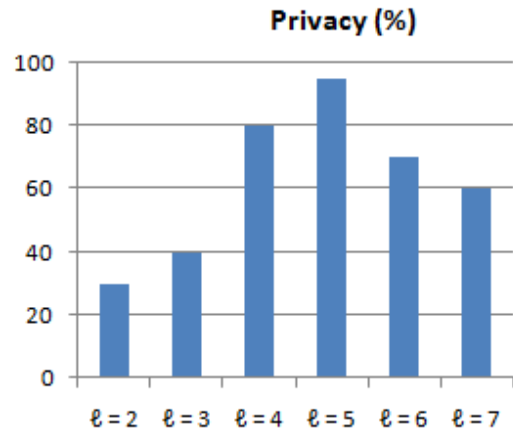


Dataset Before Slicing



Sliced Dataset

We tested the effectiveness of the approach by using different values for ℓ . We got the best results with $\ell=5$. The percentage of privacy we got, with different values of ℓ are given below,

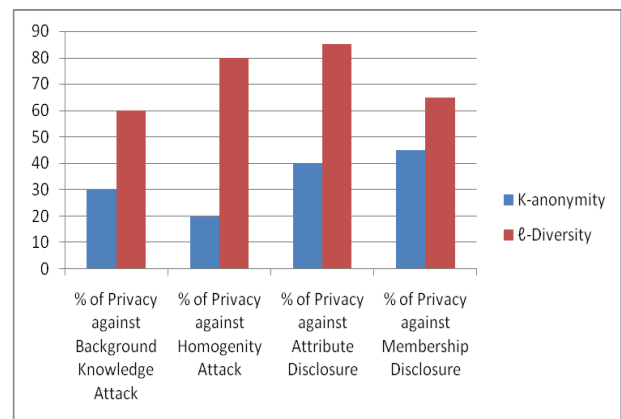


Results with different values of ℓ

Technique	High dimensional data handling	Data Utility	Privacy Protection
Generalization	Cannot Handle	Low	Low
Bucketization	Can Handle	Low	Low
K-Anonymity	Can Handle	High	Low
ℓ -Diversity	Can Handle	High	High

Table 1. Comparison of other techniques with our approach

Also Slicing prevents many privacy attacks, which ultimately enhances the privacy of sensitive information.



VII. CONCLUSION AND FUTURE SCOPE

In this way, slicing with ℓ -diversity requirement preserves the privacy of sensitive information by protecting the dataset against two privacy threats that are attribute disclosure and membership disclosure. Slicing preserves of sensitive information by breaking association across the column and preserving association within the column. The ℓ diversity approach enhances privacy preservation as it ensures the diversity among the sensitive attributes. In slicing, each column contains

much fewer attributes than the whole table, attribute partition enables slicing to handle high-dimensional data.



In future, Data mining tasks can be designed for sliced data. Also overlapping slicing can be performed which duplicates an attribute in more than one columns. This could provide better data utility.

ACKNOWLEDGMENT

This is to acknowledge and thanks to all individuals who played defining role in shaping this paper. Without their constant support, guidance and assistance this paper would not have been completed. In addition, the we would like to thank to our family members and friends for their guidance, support, encouragement and advice.

REFERENCES

- [1] Li, Ninghui Li, Jian Zhang, Ian Molloy "Slicing: A New Approach to Privacy Preserving Data Publishing" IEEE Transactions on Knowledge and Data Engineering, volume:24, Issue:3, 2012
- [2] J. Li, Y. Tao, and X. Xiao. Preservation of proximity privacy in publishing numerical sensitive data. In SIGMOD, pages 473–486, 2010
- [3] J. Brickell and V. Shmatikov. The cost of privacy: destruction of data-mining utility in anonymized data publishing. In KDD, pages 70–78, 2008
- [4] V. Rastogi, D. Suciu, and S. Hong. The boundary between privacy and utility in data publishing. In VLDB, pages 531–542
- [5] V. Rastogi, D. Suciu, and S. Hong. The boundary between privacy and utility in data publishing. In VLDB, pages 531–542, 2007.
- [6] A. Inan, M. Kantarcioglu, and E. Bertino. Using anonymized data for classification. In ICDE, 2009.
- [7] X. Xiao and Y. Tao. Anatomy: simple and effective privacy preservation. In VLDB, pages 139–150.
- [8] L. Sweeney. Achieving k-anonymity privacy protection using generalization and suppression. Int. J. Uncertain. Fuzz., 10(6):571–588
- [9] C. Aggarwal. On k-anonymity and the curse of dimensionality. In VLDB, pages 901–909

	<p>Snehal S. Kalange received the Bachelor degree (B.E.) in Information Technology in 2012 from BVCOEW, Pune. She is now pursuing Master's degree in Computer Engineering at VidyaPratishthan's College of Engineering, BARAMATI. Her current research interests include Data mining. Email: pritikalange1591@gmail.com</p>
	<p>D. M. Padulkar Graduated in Computer Science and Engineering, from Shivaji University, Kolhapur. He has completed his Post-graduate degree from Government College of Engineering Pune(CoEP). He has to his credit international conferences and journal papers. He has delivered expert lectures on Design and Analysis of Algorithms, System Programming, Computer Graphics, Compiler Construction at different engineering colleges under Shivaji University, Solapur University and Pune University. Digambar is having 10 years of experience at undergraduate and Post graduate level Email: dm.padulkar2006@gmail.com</p>