ISSN: 2319-8753
International Journal of Innovative Research in Science, Engineering and Technology
Vol. 2, Issue 4, April 2013

A New Data Anonymization Technique used For Membership Disclosure Protection

Vijay R. Sonawane[1], Kanchan S. Rahinj [2]

Associate Professor, Dept. of IT, AVCOE Sangamner, Maharashtra, India[1]

Student of M.E., Dept. of IT, AVCOE Sangamner, Maharashtra, India[2]

Abstract : **Several anonymization techniques, like generalization and bucketization, have been intended for privacy preserving microdata publishing. current work has shown that generalization loses significant amount of information, particularly for high-dimensional data. on the other hand, 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. In this paper, we present a new technique called slicing, in that data is partition into both horizontally and vertically. We demonstrate that slicing preserves better data utility than generalization and can be used for membership disclosure protection. Another main advantage of slicing is that it can handle high-dimensional data. We illustrate how slicing can be used for attribute disclosure protection and build up an efficient algorithm for computing the sliced data that obey the ℓ-diversity requirement. Our workload experiments verify that slicing preserves better utility than generalization and is more effective than bucketization in workloads involving the sensitive attribute. Our experiments also show that slicing can be used to prevent membership disclosure.**

Keywords: Generalization, Bucketization, membership disclosure protection, attribute disclosure protection.

I. .INTRODUCTION

Privacy-preserving publishing of microdata has been studied widely in current years. Microdata contain records each of which records contains information about an individual, such as a person, household, or an organization. some microdata anonymization techniques have been proposed. The most popular techniques are generalization for k-anonymity [3] and bucketization for ℓ diversity [5]. In both approaches attributes are divided into three categories: (1) some attributes are identifiers that can distinctively identify an individual, such as Name or else 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 recognize an individual, for e.g., Birth-date, Zip code and Sex; (3) some attributes are Sensitive Attributes (SAs), which are anonymous to the adversary and are considered sensitive, such the same as Disease and Salary. Mutually in generalization and bucketization, one first eliminates identifiers from the data and then partitions tuples into buckets. These two techniques differs in the next step. Generalization convert the QI-values in each bucket into "less specific but semantically consistent" values so that tuples in the same bucket cannot be differentiated by their QI value. In bucketization, one separates the SAs from the QIs by arbitrarily permuting the SA values in each bucket. That anonymized data consists of a set of buckets with permuted sensitive attribute values.

II. BACKGROUND

Generalization for k-anonymity [1] losses considerable amount of information, particularly for high-dimensional data. This is due to the following reasons. First, generalization for k-anonymity undergo from the curse of dimensionality. In order to generalization for effective records in the similar bucket must be close up to each other so that generalizing the records would not lose too much information. on the other hand, in high-dimensional data, most of data points have similar distances with each other, forcing a large amount of generalization to assure k-anonymity [1] even for relative small k's. Then the second is, in order to perform data analysis as well as data mining tasks on the generalized table, the data analyst have to make the uniform distribution assumption that every value in a generalized set is equally doable, as no more distribution assumption can be justified. This significantly decrease the data utility of the generalized data. And the Third is, because each attribute is generalized separately, correlations between two or more different attributes are missing. In order to study attribute correlations on the generalized table, the data analyst has to assume that each possible combination of attribute values is equally

ISSN: 2319-8753
International Journal of Innovative Research in Science, Engineering and Technology
Vol. 2, Issue 4, April 2013

possible. This is an natural problem of generalization that avoid effective analysis of attribute correlations.

While bucketization [4] has better data utility than generalization, also it has several limitations. First, bucketization gives attention to membership disclosure [6]. Because bucketization issues the QI values in their original forms, an opposition can find out whether an individual has a record in the published data or not. As shown in following table, some percents of the individuals in the United States can be uniquely identified using only three attributes (Birth_date, Sex, and Zip_code). A microdata (e.g., census data) usually includes many other attributes besides those three attributes. That means the membership information of most persons can be inferred from the bucketized table. Second, bucketization needed a clear separation between QIs and SAs. However, in many datasets, it is ambiguous which attributes are QIs and which are SAs. Third, separate the sensitive attribute from the QI attributes, bucketization split the attribute correlations between the QIs and the SAs.[4]

In this paper, we introduce a novel data anonymization method called slicing to improve the current state of the art. In that method called slicing the dataset partitions into both vertically and horizontally. In that vertical partitioning is done by grouping attributes into columns based on the correlations among the attributes. Each of the column contains a subset of attributes that are highly correlated. And the horizontal partitioning is done by grouping tuples into the buckets. Finally, within each bucket, values in each column are randomly sorted to break the linking between different columns. The vital idea of slicing is to break the relationship cross columns, but to preserve the relationship within each column.

III. RELATED WORK

1. Many existing clustering algorithms for e.g., k-means requires the calculation of the "centroids". But there is no perception of "centroids" in our setting where each attribute forms a data point in the clustering space.[1]
2. A k-medoid method is very robust to the existence of outliers i.e., data points that are extremely far away from the remaining data points.
3. The order in which the data points are examined does not affect the clusters calculate from the k-medoid method.[1]

A. *Disadvantages:*

1. Existing anonymization techniques or algorithms can be used for column generalization, for e.g., Mondrian. These algorithms can be applied on the sub table containing only attributes in one column to make sure the secrecy requirement.[2]
2. Existing data analysis like query answering methods can be easily used on the sliced data.
3. Existing privacy actions for membership disclosure protection include differential privacy and presence.[6]

IV. THE PROPOSED SYSTEM

This paper present a novel technique called slicing, in that data can be partitions in both horizontally and vertically. In that we illustrate that it preserves better data utility than generalization [3] and can be used for membership disclosure protection. Also it has one another important advantage that is it can handle high-dimensional data. In this paper we show that how this slicing technique can be used for attribute disclosure protection and develop an efficient algorithm for computing the sliced data that follow the ℓ-diversity requirement [1]. Our workload experiments verify that slicing preserves better utility than generalization and is more efficient than bucketization in that workloads involving the sensitive attribute.[5]

A. *Advantages*:

1. We introduce a data anonymization technique called slicing to get better current state of the art.
2. We show that slicing can be effectively used for avoiding
attribute disclosure, depending on the privacy requirement of ℓ-diversity.
3. We develop an efficient algorithm for calculate the sliced table that satisfies ℓ-diversity. This proposed algorithm partitions attributes into columns, be relevant column generalization, and partitions tuples interested in buckets.
4. We perform extensive workload experiments. Our results prove that slicing preserves much better data utility than generalization. In workload involving the sensitive attribute, slicing is more effective than bucketization.[5] In some experiments, slicing shows better performance than using the original data. Our experiments shows the limitations of bucketization in membership disclosure protection and slicing remedies these type of limitations.

B. *Slicing Algorithms:*

Our algorithm consists of some phases: that are tuple partitioning and Diversity check. Now we describe these phases.

1. *Algorithm tuple-partition(T, ℓ)*

1. Q = {T}; SB = ∅.
2. while Q is not empty
3. remove the first bucket B from Q; Q = Q − {B}.
4. split B into two buckets i.e. B1 and B2, as in Mondrian.
5. if diversity-check(T, Q ∪ {B1,B2} ∪ SB, ℓ)
6. Q = Q ∪ {B1,B2}.
7. else SB = SB ∪ {B}.
8. return SB.

ISSN: 2319-8753
International Journal of Innovative Research in Science, Engineering and Technology
Vol. 2, Issue 4, April 2013

2. Algorithm diversity-check(T,T`, ℓ)

1. for each tuple t ∈ T, L[t] = ∅.
2. for each bucket B in T`
3. record f(v) for each column value v in bucket B.
4. for each tuple t ∈ T
5. calculate p(t,B) and find D(t,B).
6. L[t] = L[t] ∪ {hp(t,B),D(t,B)i}.
7. for each tuple t ∈ T
8. calculate p(t, s) for each s based on L[t].
9. if p(t, s) ≥ 1/ℓ, return false.
10. return true.

In this section, we first give an example to demonstrate slicing. Then we formalize slicing, compare it with generalization and bucketization, and talk about privacy threats that slicing can address. Table shows an example microdata table and its anonymized versions using various anonymization techniques .Stepwise explanation is given bellow:

1. Original Data
2. Generalized Data
3. Bucketized Data
4. Multiset-based Generalization Data
5. One-attribute-per-Column Slicing Data
6. Sliced Data

1.Original Data:

We conduct the extensive workload experiments. Our results are confirm that slicing preserves much better data utility than the generalization. In these involving the sensitive attribute, slicing is also more effectual than bucketization. In some classification experiments, the novel technique i.e. slicing shows better performance than the original data.

TABLE 1: The original table

Age	Sex	Zip code	Disease
22	M	47906	dyspepsia
22	F	47906	flu
33	F	47905	flu
52	F	47905	bronchitis
54	M	47302	flu
60	M	47302	dyspepsia
60	M	47304	dyspepsia
64	F	47304	gastritis

2.Generalized Data:

In order to,perform data analysis or data mining tasks on the generalized table[2][3], the data analyst has to make the uniform distribution supposition that each value in a generalized interval is equally possible, as no additional distribution assumption can be justified. This significantly decreases the data utility of the generalized data

TABLE 2: The Generalized Table

Age	Sex	Zip code	Disease
[20-52]	*	4790*	Dyspepsia
[20-52]	*	4790*	Flu
[20-52]	*	4790*	Flu
[20-52]	*	4790*	bronchitis
[54-64]	*	4730*	Flu
[54-64]	*	4730*	Dyspepsia
[54-64]	*	4730*	Dyspepsia
[54-64]	*	4730*	gastritis

3.Bucketized Data:

In that we show the efficiency of slicing in membership disclosure protection [7][8]. For this purpose, we calculate the number of fake tuples in the sliced data. Also we compare the number of matching buckets for original tuples and that for fake tuples. Our experimental results illustrate that bucketization does not prevent membership disclosure [9] as almost every tuple is distinctively identifiable in the bucketized data [4].

TABLE 3: The Bucketized Table

Age	Sex	Zip code	Disease
22	M	47906	Flu
22	F	47906	Dyspepsia
33	F	47905	Bronchitis
52	F	47905	Flu
54	M	47302	Gastritis
60	M	47302	Flu
60	M	47304	Dyspepsia
64	F	47304	Dyspepsia

4.Multiset-based Generalization Data:

We monitor that this multiset-based generalization is the same to a trivial slicing scheme where each column contains exactly one attribute, because both approaches preserve the exact values in each attribute but split the association between them within one bucket.

TABLE 4: Multiset-Based Generalization

Age	Sex	Zip code	Disease
22:2,33:1,52:1	M:1,F:3	47905:2,47906:2	Dysp.
22:2,33:1,52:1	M:1,F:3	47905:2,47906:2	Flu
22:2,33:1,52:1	M:1,F:3	47905:2,47906:2	Flu
22:2,33:1,52:1	M:1,F:3	47905:2,47906:2	Bron.
54:1,60:2,64:1	M:3,F:1	47302:2,47304:2	Flu
54:1,60:2,64:1	M:3,F:1	47302:2,47304:2	Dysp.
54:1,60:2,64:1	M:3,F:1	47302:2,47304:2	Dysp.
54:1,60:2,64:1	M:3,F:1	47302:2,47304:2	Gast.

5.One-attribute-per-Column Slicing Data:

We observe that while one-attribute-per-column slicing preserves attribute distributional information, it does destroy attribute correlation, for the reason that each attribute is in its own column. In slicing, one groups associated attributes together in one column and save their correlation. For instance, in the sliced table shown in Table correlations between Age and Sex and correlations between Zipcode and
Disease are conserved. In fact, the sliced table encodes the same amount of information as the original data with

ISSN: 2319-8753
International Journal of Innovative Research in Science, Engineering and Technology
Vol. 2, Issue 4, April 2013

observe to correlations between attributes in the same column.

TABLE 5: One-Attribute-Per-Column Slicing

Age	Sex	Zip code	Disease
22	F	47906	Flu
22	M	47905	Flu
33	F	47906	Dysp.
52	F	47905	Bron.
54	M	47302	Dysp.
60	F	47304	Gast.
60	M	47302	Dysp.
64	M	47304	Flu

6.Sliced Data:

Another important advantage of slicing is its capability to handle high-dimensional data. By dividing attributes into columns, slicing condense the measurements of the data. Each of wich column of the table can be viewed as a sub-table with a lesser dimensionality. Slicing is also not similar from the approach of publishing multiple independent sub-tables in that these sub-tables are associated by the buckets in slicing.

TABLE 6: The Sliced Table

(Age,Sex)	(Zipcode,Disease)
(22,M)	(47905,Flu)
(22,F)	(47906,dysp.)
(33,F)	(47905,bron.)
(52,F)	(47906,flu)
(54,M)	(47304,gast.)
(60,M)	(47302,Flu)
(60,M)	(47302,dysp.)
(64,F)	(47304,dysp.)

V. CONCLUSION AND FUTURE SCOPE

This paper presents a new approach called slicing to privacy-preserving microdata distributing. It overcomes thelimitations of generalization and bucketization and preserves better utility while protecting in opposition to privacy threats. We illustrate how to apply slicing to prevent attribute disclosure and membership disclosure. Our experiments shows that slicing preserves better data utility than generalization and is more effective than bucketization. The general method proposed by this work is that: Before anonymizing the data, one can investigate the data characteristics and use these individuality in data anonymization. The basis is that one can design better data anonymization techniques when we know that the data is better .

ACKNOWLEGEMENT

We thank all sponsors in the footnote on the first page for funding this ongoing research project and all volunteers for their involving this research project. We would also like to thanks to the unidentified referees for their constructive and helpful comment.

REFERENCES

[1] C. Aggarwal "K-anonymity and the curse of dimensionality" In VLDB, pages 901–909, 2005.
[2] P. Samarati "Protecting respondent's privacy in microdata release" In KDE, 13(6):1010–1027, 2001.
[3] L. Sweeney "k-anonymity: A model for protecting privacy" International Journal Uncertain. Fuzz., 10(5):557–570, 2002.
[4] X. Xiao,Y. Tao "Anatomy: simple and effective privacy preservation" In VLDB, pages 139–150, 2006.
[5]A. Machanavajjhala, J. Gehrke, D. Kifer, M. Venkitasubramaniam "ℓ-diversity: Privacy beyond k-anonymity" In ICDE, page 24, 2006.
[6]M.E. Nergiz, M. Atzori, C. Clifton "Hiding the presence of individuals from shared databases" In SIGMOD, pages 665–676, 2007.
[7] I. Dinur, K. Nissim "Revealing information while preserving privacy" In PODS, pages 202–210, 2003.
[8]C. Dwork, F. McSherry, K. Nissim, A. Smith "Calibrating noise to sensitivity in private data analysis" In TCC, pages 265–284, 2006.

BIOGRAPHY

Mr. Sonawane Vijay R. received M.Tech degree in Computer Science & Technology from Shivaji University, Kolhapur (MS), India, in 2011.He is currently pursuing PhD in Computer Engineering from K.L.university, Vijayawada. He has published Over 6 papers in international journals and Conferences in the areas of E- Commerce And Data Mining His current area of research is Clustering and Mining Web Data

Ms. Rahinj kanchan shivaji receiving her B.E. in Information Technology from Pune University, in 2012. She is currently pursuing her M.E. in Information Technology from University of Pune.