Reliable Techniques For Data Publishing From Sparse Datasets

Dept. of Computer Science and Engineering
Kakinada Institute of Engineering & Technology, Affiliated to JNTU KAKINADA
Korangi, East .Godavari .District, Andhra. Pradesh.
1 Janipalli Lavanya, Final M.Tech Student (lavanya.janipalli@gmail.com)
2 D. Srinivas, Asst. Professor (kietdskiet@gmail.com)

Abstract -- **Confidential Information pertained to individuals must not be revealed, but a dataset can be useful for studying the characteristics of a population in adhoc analysis. Confidential Information protection is an important issue in the release of data for mining purposes. Recent studies have shown that a more sophisticated model is necessary to protect the association of individuals to sensitive information. This paper presents effectiveness and efficiency of reliable techniques (generalization, bucketization and slicing) shown by experiments. This work has great applicability in both public and private sectors that share information for mutual benefits and productivity. Generalization shown that there is a considerable loss of amount from high-dimensional data. On the other hand bucketization does not prevent membership disclosure. We discussed about a novel technique called "slicing" which preserves a better utility than generalization and bucketization through experimentations.**

Keywords— ***Slicing, Generalization,Bucketization***

1. Introduction

Consider a data holder, such as a hospital or a bank, that has a privately held collection of person-specific, field structured data. Suppose the data holder wants to share aversion of the data with researchers. How can a data holder release a version of its private data with scientific guarantees that the individuals who are the subjects of the data cannot be re-identified while the data remain practically useful? So, information about specific individuals, are increasingly becoming public in response to "open government" laws and to support data mining research. Some datasets include legally protected information such as health histories; which many people may view as private or sensitive. As though Micro-data are characterized by high dimensionality and sparsity .

Each record contains many attributes (i.e., columns in a database schema), which can be viewed as dimensions. Sparsity means that for the average record, there are no "similar" records in the multi-dimensional space defined by the attributes. Generalizations of the data, where low level or "primitive" data are replaced by higher-level concepts throw the use of concept hierarchies for these we use K-anonymity technique[]. Bucketization means group several records and mix their sensitive values for this we use l-diversity technique. The slice operation performs a selection on one dimension of the given cube, resulting in a subcube. Generalizations of the data, where low level or "primitive" data are replaced by higher-level concepts throw the use of concept hierarchies for these we use K-anonymity technique. Bucketization means group several records and mix their sensitive values for this we use l-diversity technique. In both approaches, attributes are partitioned into three categories: (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 together, can potentially identify an individual,e.g.,Birth- the adversary may already know (possibly from other publicly-available databases) and which, when taken together, can potentially identify an individual, e.g., Birth- date, 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.

2. Related Work:

2.1 Basic Definitions

2.1.1 Quasi-Identifier Attribute Set A quasi-identifer is a minimal set of attributesX1; :::;Xd in table T that can be joined with external information to re-identify individual records[2].

2.1.2 K-Anonymity Property is k-anonymous with respect to attributes X1; :::;Xd if every unique tuple (x1; :::; xd) in the (multiset) projection of T on X1; :::;Xd occurs at least k times. That is, the size of each equivalence
class in T with respect to X1; :::;Xd is at least k[2].

2.1.3 Sensitive Attributes: In many real-world scenarios, an individual may have several sensitive values. Ex: diseases[3].

2.1.4 Generalization

Let V be the domain of an attribute t.V . A generalization W of V is a new domain formed by partitioning V into disjoint buckets and identifying all the points in a bucket with one value in W. A generalization map is a function _ : V ! W such that _(v) corresponds to the bucket that contains v[1].

2.2 Example:

In the table the three QI attributes are{Age,Sex,Zipcode}, and the sensitive attribute(SA)isdisease.The Quasi-Identifying attributes are set of attributes that can be linked with public available data sets to reveal personal identity

Age	Sex	Zipcode	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

(a) The original table

Generalizations of the data, where low level or "primitive" data are replaced by higher-level concepts throw the use of concept hierarchies for these we use K-anonymity technique. The generalized shown that satisfies 4-anonymity where each attribute value is replaced with the multiset of values in the bucket

Age	Sex	Zipcode	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

(b) The generalized table

Bucketization means group several records and mix their sensitive values for this we use l-diversity technique. This table shows the values which satisfies 2-Diversity. Within each bucket, values in each column are randomly permuted to break the linking between different columns.

Age	Sex	Zipcode	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

(c) The bucketized table

One problem with generalization is that it cannot handle high- dimensional data due to "the curse of dimensionality. Bucketization was proposed to remedy this drawback. The bucketization method 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 bucketized data consists of a set of buckets with permuted sensitive attribute values. Finally, another widely-used method is suppression which replaces a QI value by a '*' character. Each attribute is generalized separately; correlations between different attributes are lost. In order to study attribute correlations on the generalized table, the data analyst has to assume that every possible combination of attribute values is

equally possible. This is an inherent problem of generalization that prevents effective analysis of attribute correlations.

3. Drive for Slicing

In this paper, we discuss a novel data anonymization technique called slicing to improve the current state of the art. 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 diff erent columns. The basic idea of slicing is to break the association cross columns, but to preserve the association within each col- umn. This reduces the dimensionality of the data and pre- serves 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 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 is an effective technique for membership disclosure protection

4. Problem Formulation

Let T be the microdata table to be published. T contains d attributes: A = {A1,A2, . . . ,Ad} and their attribute domains are {D[A1],D[A2], . . . ,D[Ad]}. A tuple t ∈ T can be represented as t = (t[A1], t[A2], ..., t[Ad]) where t[Ai] ($1 \leq i \leq d$) is the Ai value of t[10].

4.1 Attribute partition and columns.

An attribute partition consists of several subsets of A, such that each attribute belongs to exactly one subset. Each subset of attributes is called a column. Specifically, let there be c columns C1,C2, . . . ,Cc, then $\cup_{i=1}^{c} Ci = A$ and for any $1 \leq i1 \neq i2 \leq c$, $Ci1 \cap Ci2 = \emptyset$ [10].

4.2 Tuple partition and buckets.

A tuple partition consists of several subsets of T, such that each tuple belongs to exactly one subset. Each subset of tuples is called a bucket. Specifically, let there be b buckets B1,B2, . . . ,Bb, then $\cup_{i=1}^{b} Bi = T$ and for any $1 \leq i1 \neq i2 \leq b$, $Bi1 \cap Bi2 = \emptyset$.[10]

4.3 Slicing.

Given a microdata table T, a slicing of T is given by an attribute partition and a tuple partition[4].

4.4 Column Generalization

Given a microdata table T and a column Ci = {Ai1,Ai2, . . . ,Aij}, a column generalization for Ci is defined as a set of non-overlapping j-dimensional regions that completely cover $D[Ai1] \times D[Ai2] \times \ldots \times D[Aij]$. A column generalization maps each value of Ci to the region in which the value is contained.

Matching Buckets

Let {C1,C2, . . . ,Cc} be the c columns of a sliced table. Let t be a tuple, and t[Ci] be the Ci value of t. Let B be a bucket in the sliced table, and B[Ci] be the multiset of Ci values in B. We say that B is a matching bucket of t iff for all $1 \leq i \leq c$, $t[Ci] \in B[Ci]$.

4.5Algorithms:

4.5.1Alogorithm tuple-partition(T,l)

1.Q={T}; SB=ϕ.
2.while Q is not empty
3.remove the first bucket B from Q;Q=Q-{B}
4.split B into two buckets B1 and B2,as in Mondrian
5.if diversity-check(T,QU{B1,B2}U SB,ℓ).

6.Q=Q U {B1,B2}.
7.else SB=SB U {B}
8.return SB.

4.5.2Algorithmdiversity-check(T,T ,ℓ)*

1.for each tuple t € T,L(t)=ϕ
2.for each bucket B in T*.
3.record f(v)for each column value in bucket B.
4.for each tuple t € T
5.calculate p(t,B) and find D(t,B).
6.L[t]=L[t] U {(p(t,B),D(t,B))}.
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 flase.
10.return true.

4.6 Slicing versus Generalization

I. There are several types of recodings for generalization which preserves the information is local recoding.

II. Same attribute value may be generalized differently when they appear in different buckets.

4.7 Slicing versus Bucketization

III. Bucketization can be viewed as a special case of slicing. Where there are exactly 2 columns : 1 column contains only the SA, and the other contains all the QIs.

IV. By partitioning attributes into more than 2 columns, slicing can be used to prevent membership disclosure

V. Bucketization, which requires clear separation of QI attributes and the sensitive attribute, slicing can be used without such a separation.

4. Experimental Example:

A generiled table where each attribute value is replaced with the multiset of values in the bucket shown in table 1(d). Table 1(d) is the result of using mul- tisets of exact values rather than generalized values. For the Age attribute of the first bucket, we use the multiset of exact values {22,22,33,52} rather than the generalized interval [22 − 52]. The multiset of exact values provides more information about the distribution of values in each attribute than the generalized interval. Therefore, using multisets of exact values preserves more information than generalization.

Age	Sex	Zipcode	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.

(d) Multiset-based generalization

Age	Sex	Zipcode	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

(e) One-attribute-per-column slicing

Table 1(e) is equivalent to Table 1(d). Now comparing Table 1(e) with the sliced table shown in Table 1(f), we observe that while one-attribute-per-column slicing preserves attribute distributional information, it does not preserve attribute correlation, because each attribute is in its own column. In slicing, one groups correlated attributes together in one column and preserves their correlation. For example, in the sliced table shown in Table 1(f), correlations between Age and Sex and correlations between Zipcode and Disease are preserved. In fact, the sliced table encodes the same amount of information as the original data with regard to correlations between attributes in the same column.

(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.)

(f) The sliced table

6.Conclusion and Future Work

The slicing technique is an effiecient for privacy preserving micro data publishing.it overcomes the limitations of Generealization and Bucktezition against privacy threats.our expermient shows that slicing preserves better data utility than other techniques. Here we consider slicing where each attribute exactly one column the extesion work is notion of overlapping slicing which duplicates an attribute in more than one column.

7. References:

1. D. Kifer and J. Gehrke. Injecting utility into anonymized datasets. In SIGMOD, pages 217–228,2006.
2.Mondrian Multidimensional K-Anonymity Kristen LeFevre David J. DeWitt Raghu Ramakrishnan
3.University of Wisconsin, MadisonK. LeFevre, D. J. DeWitt, and R. Ramakrishnan.Workload-aware anonymization. In KDD, pages 277–286, 2006.
4.T. Li and N. Li. Injector: Mining background knowledge for data anonymization. In ICDE, pages 446–455, 2008.
5.T. Li and N. Li. On the tradeoff between privacy and utility in data publishing. In KDD, pages 517–526, 2009
6.M. E. Nergiz, M. Atzori, and C. Clifton. Hiding the presence of individuals from shared databases. In SIGMOD, pages 665–676, 2007.
7.Y. Xu, K. Wang, A. W.-C. Fu, and P. S. Yu.Anonymizing transaction databases for publication. In KDD, pages 767–775, 2008.
8.B. C. M. Fung, K. Wang, and P. S. Yu. Top-down specialization for information and privacy preservation. In ICDE, pages 205–216, 2005.
9.B.-C. Chen, R. Ramakrishnan, and K. LeFevre.Privacy skyline: Privacy with multidimensional adversarial knowledge. In VLDB, pages 770–781, 2007.
10.Slicing: A New Approach to Privacy Preserving Data Publishing Tiancheng Li, Ninghui Li, Jian Zhang, Ian Molloy Purdue University, West Lafayette, IN 47907 {li83,ninghui}@cs.purdue.edu, jianzhan@purdue.edu,molloy@cs.purdue.ed u

Miss J Lavanya is a student of Kakinada Institute of Engineering &Technology (KIET), Korangi. Currently She is pursuing his M.Tech (CSE) (11B21D5816) From this College. Her Areas of interest are Data Mining.

Mr. D.Srinivas is working as Assistant Professor in KIET. He has 6 years of Teaching experience. He completed his B.tech from KIET in 2007.He completed his M.Tech from GIET Rajahmundry in 2010. His Areas of interests are DBMS & Networks He had Published his paper in International Journal of computer science & Technology.

