International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2019 IJSRCSEIT | Volume 5 | Issue 4 | ISSN : 2456-3307
DOI : https://doi.org/10.32628/CSEIT1953193

Secure Data Protection Using Slicing as a Confusion Technique

V Veda Sahithi*, V Swarna Kamalam
Information Technology Department, JNTUH University/Bhoj Reddy Engineering College for Women, Hyderabad, Telangana, India

ABSTRACT

Data Mining deals with automatic extraction of previously unknown patterns from large amounts of data sets. These data sets typically contain sensitive individual information or critical business information, which consequently get exposed to the other parties during Data Mining activities. Secure data protection has been one of the greater concerns in data mining. Several anonymization techniques, such as generalization and bucketization, have been designed for privacy protective microdata publishing. The 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. Solution to this problem is provided by we introduce a novel data anonymization technique called slicing to improve the current state of the art.

Keywords : Data Mining, Privacy Protection, Data anonymization, Security, L diversity.

I. INTRODUCTION

In modern days' organizations are extremely dependent on Data Mining results to provide better service, achieving greater profit, and better decision-making. For these purposes organizations collect huge amount of data. This data includes sensitive data about Individuals or organizations. While running Data Mining algorithm against such data, the algorithm not only extracts the knowledge but it also reveals the information which is considered to be private. The real threat is that once information gets exposed to unauthorized party, it will be impractical to stop misuse. Privacy can for instance be threatened when Data Mining techniques uses the identifiers which themselves are not very sensitive, but are used to connect personal identifiers such as addresses, names etc., with other more sensitive personal information. Security is very important for trusted collaboration and interactions. Because of these privacy and data security concerns in data mining, the data owner hesitates while sharing data for data mining activities. And this creates obstacle in data mining task. Secure data protection techniques give new direction to solve this problem. The 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. 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 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 Zip

code. 3) some attributes are Sensitive Attributes (SAs), which are unknown to the adversary and are considered sensitive, such as Disease and Salary.

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

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

Table 1 Generalization

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

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

Table 2 Bucketization

Privacy- protection publishing of microdata has been studied extensively in recent years. Microdata contains records each of which contains information about an 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 identifiers that can uniquely identify an individual, such as Name or Social Security Number.

2) Some attributes are Quasi Identifiers (QI), which the opponent may already know (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 opponent and are considered sensitive, such as Disease and Salary.

II. METHODS AND MATERIAL

In both generalization and bucketization, one first removes identifiers from the data and then partitions tuples into buckets.

The two techniques differ in the next step.

• 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.

• Bucketization, one separates the SAs from the QIs by randomly permuting the SA values in each bucket. The anonymized data consist of a set of buckets with permuted sensitive attribute values.

It has been shown that generalization for k-anonymity losses considerable amount of information, especially for high-dimensional data. This is due to the following three reasons. 1. First, generalization for k-anonymity suffers from the curse of dimensionality. In order for generalization to be effective, records in the same bucket must be close to each other so that generalizing the records would not lose too much information. However, in high dimensional data, most data points have similar distances with each other, forcing a great amount of generalization to satisfy k-anonymity even for relatively small k's. 2. Second, in order to perform data analysis or data mining tasks on the generalized table, the data analyst has to make the uniform distribution assumption that every value in a generalized interval/set is equally possible, as no other distribution assumption can be justified. This significantly reduces the data utility of the generalized data. 3. Third, because 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. 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 (Birthdate, Sex, and Zipcode). A microdata (e.g., census 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.

III. RESULTS AND DISCUSSION

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. The single column that needs protection and should not be disclosed to users when they query the table.

1. Attribute partitioning:

・It 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 association of uncorrealated attribute values is much less frequent and thus more identifiable.

・Therefore, it is better to break associations between uncorrelated attributes, inorder to protect privacy.

2.Tuple partitioning:

・Vertical partitioning is done by grouping attributes into columns based on the correlations among the attributes .In the tuple partitioning phase, tuples are partitioned into buckets. The algorithm maintains two data structures:

1) a queue 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 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. 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

Table 3 Slicing

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)

a tuple t ={v1,v2,...,vc} , 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), vi 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.

Slicing:

・First, we introduce slicing as a new technique for privacy preserving data publishing. Slicing has several advantages when compared with generalization and bucketization. It preserves better data utility than generalization. It preserves more attribute correlations with the SAs than bucketization. It can also handle high-dimensional data and data without a clear separation of QIs and SAs.

・Second, we show that slicing can be effectively used for preventing attribute disclosure, based on the privacy requirement of 'L-diversity'. We introduce a notion called 'diverse slicing, which ensures that the adversary cannot learn the sensitive value of any individual with a probability greater than 1/L.

・Third, we develop an efficient algorithm for computing the sliced table that satisfies L diversity. Our algorithm partitions attributes into columns, applies column generalization, and partitions tuples into buckets. Attributes that are highly correlated are in the same column; this preserves the correlations between such attributes. The associations between uncorrelated attributes are broken, this provides better privacy as the associations between such attributes are less frequent and potentially identifying.

・ Fourth, we describe the intuition behind membership disclosure and explain how slicing prevents membership disclosure. A bucket of size k can potentially match kc tuples where c is the number of columns. Because only k of the k^c tuples are actually in the original data, the existence of the other k^c-k tuples hides the membership information of tuples in the original data. Finally, we conduct extensive workload experiments. Our results confirm that slicing preserves much better data utility than generalization. In workloads involving the sensitive attribute, slicing is also more effective than bucketization. Our experiments also show the limitations of bucketization in membership disclosure protection and slicing remedies these limitations.

IV. CONCLUSION

Slicing overcomes the limitations of generalization and bucketization and preserves better utility while protecting against privacy threats. We illustrate how to use slicing to prevent attribute disclosure and membership disclosure. Our experiments show that slicing preserves better data utility than generalization and is more effective than bucketization in workloads involving the sensitive attribute. The general methodology proposed by this work is that before anonymizing the data, one can analyse the data characteristics and use these characteristics in data anonymization. The rationale is that one can design better data anonymization techniques when we know the data better.

V. REFERENCES

[1] C. Aggarwal, "On k-Anonymity and the Curse of Dimensionality," Proc. Int'l Conf. Very Large Data Bases (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 Int'l Conf. 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.

Cite this article as :

V Veda Sahithi, V Swarna Kamalam, "Secure Data Protection Using Slicing as a Confusion Technique ", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 5 Issue 4, pp. 297-301, July-August 2019. Available at doi : https://doi.org/10.32628/CSEIT1953193
Journal URL : http://ijsrcseit.com/CSEIT1953193