IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

Privacy Preserving Data Publishing: Using Overlapping Slicing and Attribute Partitioning

Lavanya[*1], Dr.Sugumar R[2], M.Rajasekar[3]

[*1,2,3]Dept of CSE Vel Tech Multitech Dr.Rangarajan Dr.Sakunthala Engg College, India

lavanya.karuppiah@gmail.com

Abstract

Privacy preserving publishing is the kind of techniques to apply privacy to collected vast amount of data. The data publication processes are today still very difficult. Data often contains personally identifiable information and therefore releasing such data may result in privacy breaches; this is the case for the examples of microdata, e.g., census data and medical data. The proposed techniques in this project accelerate accessing speed of user as well as applying privacy to collected data. Several anonymization techniques were designed for privacy preserving data publishing. Recent work in data publishing has shown that generalization losses considerable amount of information, especially for high dimensional data. Bucketization, on the other hand, does not prevent membership disclosure. I propose an overlapping slicing method for handling high-dimensional data. By partitioning attributes into more than one column, we protect privacy by breaking the association of uncorrelated attributes and preserve data utility by preserving the association between highly correlated attributes. This technique releases more attribute correlations thereby, overlapping slicing preserves better data utility than generalization and is more effective than bucketization in workloads involving the sensitive attribute.

Keywords: Data Publishing, Microdata, Generalization, Bucketization, Anonymization Technique.

Introduction

The collection of digital information by governments, corporations, and individuals has created tremendous opportunities for knowledge-based decision making. There is a demand for the exchange and publication of data among various parties to achieve mutual benefits and follow regulations for publishing data. Government agencies and other organizations often need to publish micro data, e.g., medical data or census data, for research and other purposes. Typically, such data are stored in a table, and each record (row) corresponds to one individual. Each record has a number of attributes, which can be divided into the following three categories: 1) Attributes that clearly identify individuals. These are known as explicit identifiers and include, e.g., Social Security Number. 2) Attributes whose values when taken together can potentially identify an individual. These are known as quasi-identifiers, and may include, e.g., Zip code, Birth-date, and Gender. 3) Attributes that are considered sensitive, such as Disease and Salary.

When releasing micro data, it is necessary to prevent the sensitive information of the individuals from being disclosed. Two types of information disclosure have been identified in the literature. Identity disclosure and attribute disclosure. Identity disclosure occurs when an individual is linked to a particular record in the released table.

Attribute disclosure occurs when new information about some individuals is revealed, i.e., the released data make it possible to infer the characteristics of an individual more accurately than it would be possible before the data release. Identity disclosure often leads to attribute disclosure. Once there is identity disclosure, an individual is re-identified and the corresponding sensitive values are revealed. Attribute disclosure can occur with or without identity disclosure. It has been recognized that even disclosure of false attribute information may cause harm. An observer of a released table may incorrectly perceive that an individual's sensitive attribute takes a particular value and behaves accordingly based on the perception. This can harm the individual, even if the perception is incorrect.

While the released table gives useful information to researchers, it presents disclosure risk to the individuals whose data are in the table. Therefore, our objective is to limit the disclosure risk to an acceptable level while maximizing the benefit. This is achieved by anonym zing the data before release. The first step of anonymization is to remove explicit identifiers. However, this is not enough, as an adversary may already know the quasi-identifier

values of some individuals in the table. This knowledge can be either from personal knowledge (e.g., knowing a particular individual in person), or from other publicly available databases (e.g., a voter registration list) that include both explicit identifiers and quasi-identifiers.

A common anonymization approach is generalization, which replaces quasi-identifier values with values that are less-specific but semantically consistent. As a result, more records will have the same set of quasi-identifier values. We define an equivalence class of an anonym zed table to be a set of records that have the same values for the quasi-identifiers.

To effectively limit disclosure, we need to measure the disclosure risk of an anonymized table. To this end, Samarati and Sweeney introduced k-anonymity as the property that each record is indistinguishable with at least k-1 other records with respect to the quasi-identifier. In other words, k-anonymity requires that each equivalence class contains at least k records.

Related Work

Consider microdata such as census data and medical data. Typically, microdata are stored in a table, and each record (row) corresponds to one individual. Each record has a number of attributes, which can be divided into the following three categories:

1. Identifier: Identifiers are attributes that clearly identify individuals. Examples include Social Security Number and Name.
2. Quasi-Identifier: Quasi-identifiers are attributes whose values when taken together can potentially identify an individual. Examples include Zip-code, Birthdate, and Gender. An adversary may already know the QI values of some individuals in the data. This knowledge can be either from personal contact or from other publicly available databases (e.g., a voter registration list) that include both explicit identifiers and quasi-identifiers.
3. Sensitive Attribute: Sensitive attributes are attributes whose values should not be associated with an individual by the adversary. Examples include Disease and Salary.

An example of microdata table is shown in Table 2.1. As in most previous work, assume that each attribute in the microdata is associated with one of the above three attribute types and attribute types can be specified by the data publisher.

Age	Gender	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

Table 2.1: Original Table

Generalization: The generalization mechanism produces a release candidate by generalizing (coarsening) some attribute values in the original table. The basic idea is that, after generalizing some attribute values, some records would become identical when projected on the set of quasi-identifier (QI) attributes (e.g., age, gender, zip-code).Each group of records that have identical QI attribute values is called an equivalence class.

Suppression: The suppression mechanism produces a release candidate by replacing some attribute values (or parts of attribute values) by a special symbol that indicates that the value has been suppressed (e.g., "*" or "Any"). Suppression can be thought of as a special kind of generalization. For example, in Table 2.2, we can say that some digits of zip codes and all the gender values have been suppressed.

Swapping: The swapping mechanism produces a release candidate by swapping some attribute values. For example, after removing the names, the data publisher may swap the age values or swap the gender values of the patients, and so on.

Age	Gender	Zip-Code	Disease
[22-64]	*	47***	dyspepsia
[22-64]	*	47***	flu
[22-64]	*	47***	flu
[22-64]	*	47***	bronchitis
[22-64]	*	47***	flu
[22-64]	*	47***	dyspepsia
[22-64]	*	47***	dyspepsia
[22-64]	*	47***	gastritis

Table 2.2: Generalization

Bucketization: The bucketization mechanism produces a release candidate by first partitioning the original data table into non-overlapping groups (or buckets) and then, for each group, releasing its projection on the non-sensitive attributes and also its projection on the sensitive attribute. Table 2.3 is a release candidate of the bucketization mechanism when applied to Table 2.1. In this case the Condition attribute is considered to be sensitive and the other attributes are not.

The idea is that after bucketization, the sensitive attribute value of an individual would be indistinguishable from that of any other individual in the same group. Each group is also called an equivalence class.

Age	Gender	Zip-Code	Disease
[22-52]	*	4790*	dyspepsia
[22-52]	*	4790*	flu
[22-52]	*	4790*	flu
[22-52]	*	4790*	bronchitis
[54-64]	*	4730*	flu
[54-64]	*	4730*	dyspepsia
[54-64]	*	4730*	dyspepsia
[54-64]	*	4730*	gastritis

Table 2.3: Bucketization

Randomization: A release candidate of the randomization mechanism is generated by adding random noise to the data. The sanitized data could be sampled from a probability distribution (in which case it is known as synthetic data) or the sanitized data could be created by randomly perturbing the attribute values.

For example, Table 2.3 is such a release candidate for Table 2.1, where random noise is added to each attribute value. We add Gaussian noise with mean 0 and variance 4 to age and also Gaussian noise with 0 mean and variance 500 to zip code. For gender, nationality, and condition, with probability 1/4, we replace the original attribute value with a random value in the domain; otherwise, we keep the original attribute value. Note that, in general, we may add different amounts of noise to different records and different attributes.

Several application scenarios of randomization can be distinguished. In input randomization, the data publisher adds random noise to the original data set and releases the resulting randomized data, like Table 2.3.

In output randomization, data users submit queries to the data publisher and the publisher releases randomized query results. In local randomization, individuals (who contribute their data to the data publisher) randomize their own data before giving their data to the publisher. In this last scenario, the data publisher is no longer required to be trusted.

Multi-set based Generalization: Microaggregation first group's records into small aggregates containing at least *k* records in each aggregate and publishes the centroid of each aggregate. Clustering records into group of size at least k and releasing summary statistics for each cluster. Each group of records is then generalized to the same record locally to minimize information loss.

The similarity between spatial indexing and k- anonymity are observed and proposed to use spatial indexing techniques to anonymize datasets. Heuristics are presented for anonymizing one-dimensional data (i.e., the quasi-identifier contains only one attribute) and an anonymization algorithm that runs in linear time.

Multi-dimensional data is transformed to one-dimensional data using space mapping techniques before applying the algorithm for one-dimensional data. The multi-set based generalization process result is shown in Table 2.4.

Age	Gender	Zip-Code	Disease
22:2,33:1, 52:1	M:1,F: 3	47906:2,40905 :2	dyspepsi a
22:2,33:1, 52:1	M:1,F: 3	47906:2,40905 :2	flu
22:2,33:1, 52:1	M:1,F: 3	47906:2,40905 :2	flu
22:2,33:1, 52:1	M:1,F: 3	47906:2,40905 :2	bronchiti s
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	dyspepsi a
54:1,60:2, 64:1	M:3,F: 1	47302:2,47304 :2	dyspepsi a
54:1,60:2, 64:1	M:3,F: 1	47302:2,47304 :2	gastritis

Table 2.4: Multi-set based Generalization

Overlap Slicing: Overlapping slicing first partitions attributes into columns. Each column contains a subset of attributes. This vertically partitions the table. Overlapping slicing also partitions the tuples into buckets. Each bucket contains a subset of tuples. This horizontally partitions the table. Within each bucket, values in each column are randomly permutated to break the

linking between different columns. The overlapping slicing process result is shown in Table 2.5.

(Age, Gender,Disease)	(Zip-Code, Disease)
(22,M,flu)	(47906, flu)
(22,F,bronchitis)	(47906, bronchitis)
(33,F,dyspepsia)	(47905,dyspepsia)
(52,F,flu)	(47905,flu)
(54,M,dyspepsia)	(47302,dyspepsia)
(60,M,gastritis)	(47302,gastritis)
(60,M,flu)	(47304,flu)
(64,F,dyspepsia)	(47304,dyspepsia)

Table 2.5: Overlap Slicing

Information Disclosure Risks

When releasing microdata, it is necessary to prevent the sensitive information of the individuals from being disclosed. Three types of information disclosure have been identified in the literature: membership disclosure, identity disclosure, and attribute disclosure.

***Membership Disclosure*:** When the data to be published is selected from a larger population and the selection criteria are sensitive (e.g., when publishing datasets about diabetes patients for research purposes), it is important to prevent an adversary from learning whether an individual's record is in the data or not.

***Identity Disclosure*:** Identity disclosure (also called re-identification) occurs when an individual is linked to a particular record in the released data. Identity disclosure is what the society views as the clearest form of privacy violation. If one is able to correctly identify one individual's record from supposedly anonymized data, then people agree that privacy is violated. In fact, most publicized privacy attacks are due to identity disclosure.

In the case of GIC medical database, Sweeney re-identified the medical record of the state governor of Massachusetts. In the case of AOL search data, the journalist from New York Times linked AOL searcher NO. 4417749 to Thelma Arnold, a 62-year-old widow living in Lilburn, GA. And in the case of Netflix prize data, researchers demonstrated that an adversary with a little bit of knowledge about an individual subscriber can easily identify this subscriber's record in the data. When identity disclosure occurs, also say "anonymity" is broken.

Attribute Disclosure: Attribute disclosure occurs when new information about some individuals is revealed, i.e., the released data makes it possible to infer the characteristics of an individual more accurately than it would be possible before the data release. Identity disclosure often leads to attribute disclosure. Once there is identity disclosure, an individual is re-identified and the corresponding sensitive values are revealed. Attribute disclosure can occur with or without identity disclosure. It has been recognized that even disclosure of false attribute information may cause harm. An observer of the released data may incorrectly perceive that an individual's sensitive attribute takes a particular value, and behave accordingly based on the perception. This can harm the individual, even if the perception is incorrect.

In some scenarios, the adversary is assumed to know who is and who is not in the data, i.e., the membership information of individuals in the data. The adversary tries to learn additional sensitive information about the individuals. In these scenarios, our main focus is to provide identity disclosure protection and attribute disclosure protection. In other scenarios where membership information is assumed to be unknown to the adversary membership disclosure should be prevented. Protection against membership disclosure also helps to protect against identity disclosure and attribute disclosure: it is in general hard to learn sensitive information about an individual if you don't even know whether this individual's record is in the data or not.

System Architecture

A system architecture or systems architecture is the conceptual design that defines the structure and/or behavior of a system. An architecture description is a formal description of a system, organized in a way that supports reasoning about the structural properties of the system. It defines the system components or building blocks and provides a plan from which products can be procured, and systems developed, that will work together to implement the overall system. This may enable one to manage investment in a way that meets business needs.

The fundamental system organization embodied its components, their relationships to each other and the environment, and the principles governing its design and evolution. The composite of the design architectures for products and their life cycle processes. Representation of a system in which there is a mapping of functionality onto hardware and software components, is a mapping of the software architecture onto the hardware architecture, and human interaction with these components. An allocated arrangement of physical elements which provides the design solution for a consumer product or life-cycle process intended to satisfy the

requirements of the functional architecture and the requirements baseline.

Architecture is the most important, pervasive, top-level, strategic inventions, decisions, and their associated rationales about the overall structure (i.e., essential elements and their relationships) and associated characteristics and behavior.

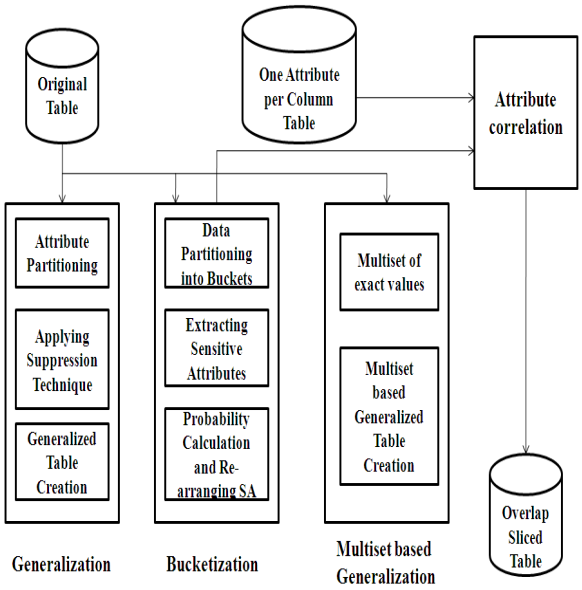

Fig 4.1: System Architecture

Conclusion

Overlap-slicing has the ability to handle high-dimensional data. By partitioning attributes into columns, overlap-slicing reduces the dimensionality of the data. Each column of the table can be viewed as a sub-table with a lower dimensionality. Overlap-slicing is also different from the approach of publishing multiple independent sub-tables in that these sub-tables are linked by the buckets in overlap-slicing. Overlap-slicing can be used without such a separation of QI attribute and sensitive attribute. A nice property of overlap-slicing is that in overlap-slicing, a tuple can potentially match multiple buckets, i.e., each tuple can have more than one matching buckets.

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] I. Dinur and K. Nissim, "Revealing Information while Preserving Privacy," Proc. ACM Symp. Principles of Database Systems (PODS), pp. 202-210, 2003.

[6] 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.

[7] B.C.M. Fung, K. Wang, and P.S. Yu, "Top-Down Specialization for Information and Privacy Preservation," Proc. Int'l Conf. Data Eng. (ICDE), pp. 205-216, 2005.

[8] G. Ghinita, Y. Tao, and P. Kalnis, "On the Anonymization of Sparse High-Dimensional Data," Proc. IEEE 24th Int'l Conf. Data Eng. (ICDE), pp. 715-724, 2008.

[9] Y. He and J. Naughton, "Anonymization of Set-Valued Data via Top-Down, Local Generalization," Proc. Int'l Conf. Very Large Data Bases (VLDB), pp. 934-945, 2009.

[10] A. Inan, M. Kantarcioglu, and E. Bertino, "Using Anonymized Data for Classification," Proc. IEEE 25th Int'l Conf. Data Eng. (ICDE), pp. 429-440, 2009.

[11] L. Kaufman and P. Rousueeuw, "Finding Groups in Data: An Introduction to Cluster Analysis," John Wiley & Sons, 1990.

[12] D. Kifer and J. Gehrke, "Injecting Utility into Anonymized Data Sets," Proc. ACM SIGMOD Int'l Conf. Management of Data (SIGMOD), pp. 217-228, 2006.

[13] N. Koudas, D. Srivastava, T. Yu, and Q. Zhang, "Aggregate Query Answering on Anonymized Tables," Proc. IEEE 23rd Int'l Conf. Data Eng. (ICDE), pp. 116-125, 2007.

[14] K. LeFevre, D. DeWitt, and R. Ramakrishnan, "Incognito: Efficient Full-Domain k-Anonymity," Proc. ACM SIGMOD Int'l Conf. Management of Data (SIGMOD), pp. 49-60, 2005.

[15] K. LeFevre, D. DeWitt, and R. Ramakrishnan, "Mondrian Multidimensional k-Anonymity," Proc. Int'l Conf. Data Eng. (ICDE), p. 25, 2006.

[16] K. LeFevre, D. DeWitt, and R. Ramakrishnan, "Workload-Aware Anonymization," Proc. ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), pp. 277-286, 2006.

[17] N. Li, T. Li, and S. Venkatasubramanian, "t-Closeness: Privacy Beyond k-Anonymity and *l*-Diversity," Proc. IEEE 23rd Int'l Conf. Data Eng. (ICDE), pp. 106-115, 2007.

[18] T. Li and N. Li, "Injector: Mining Background Knowledge for Data Anonymization," Proc. IEEE 24th Int'l Conf. Data Eng. (ICDE), pp. 446-455, 2008.

[19] T. Li and N. Li, "On the Tradeoff between Privacy and Utility in Data Publishing," Proc. ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), pp. 517-526, 2009.

[20] T. Li, N. Li, and J. Zhang, "Modeling and Integrating Background Knowledge in Data Anonymization," Proc. IEEE 25th Int'l Conf. Data Eng. (ICDE), pp. 6-17, 2009.

[21] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam, "*l*-Diversity: Privacy Beyond k-Anonymity," Proc. Int'l Conf. Data Eng. (ICDE), p. 24, 2006.

[22] D.J. Martin, D. Kifer, A. Machanavajjhala, J. Gehrke, and J.Y. Halpern, "Worst-Case Background Knowledge for Privacy-Preserving Data Publishing," Proc. IEEE 23rd Int'l Conf. Data Eng. (ICDE), pp. 126-135, 2007.

[23] M.E. Nergiz, M. Atzori, and C. Clifton, "Hiding the Presence of Individuals from Shared Databases," Proc. ACM SIGMOD Int'l Conf. Management of Data (SIGMOD), pp. 665-676, 2007.

[24] P. Samarati, "Protecting Respondent's Privacy in Microdata Release," IEEE Trans. Knowledge and Data Eng., vol. 13, no. 6, pp. 1010-1027, Nov./Dec. 2001.

[25] L. Sweeney, "Achieving k-Anonymity Privacy Protection Using Generalization and Suppression," Int'l J. Uncertainty Fuzziness and Knowledge-Based Systems, vol. 10, no. 6, pp. 571-588, 2002.

[26] L. Sweeney, "k-Anonymity: A Model for Protecting Privacy," Int'l J. Uncertainty Fuzziness and Knowledge-Based Systems, vol. 10, no. 5, pp. 557-570, 2002.

[27] M. Terrovitis, N. Mamoulis, and P. Kalnis, "Privacy-Preserving Anonymization of Set-Valued Data," Proc. Int'l Conf. Very Large Data Bases (VLDB), pp. 115-125, 2008.

[28] R.C.-W. Wong, A.W.-C. Fu, K. Wang, and J. Pei, "Minimality Attack in Privacy Preserving Data Publishing," Proc. Int'l Conf. Very Large Data Bases (VLDB), pp. 543-554, 2007.

[29] R.C.-W. Wong, J. Li, A.W.-C. Fu, and K. Wang, "(*l*, k)-Anonymity: An Enhanced k-Anonymity Model for Privacy Preserving Data Publishing," Proc. ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), pp. 754-759, 2006.

[30] X. Xiao and Y. Tao, "Anatomy: Simple and Effective Privacy Preservation," Proc. Int'l Conf. Very Large Data Bases (VLDB), pp. 139-150, 2006.

[31] Y. Xu, K. Wang, A.W.-C. Fu, and P.S. Yu, "Anonymizing Transaction Databases for Publication," Proc. ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), pp. 767-775, 2008.

[32] J. Xu, W. Wang, J. Pei, X. Wang, B. Shi, and A.W.-C. Fu, "Utility- Based Anonymization Using Local Recoding," Proc. 12th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), pp. 785-790, 2006.

.