International Journal of Modern Engineering Research (IJMER)
www.ijmer.com Vol. 3, Issue. 3, May.-June. 2013 pp-1756-1762 ISSN: 2249-6645

Improving the implementation of new approach for Data Privacy Preserving in Data Mining using slicing

Ravindra S. Wanjari Prof. Devi Kalpana

P.G. Scholar Vivekanand Institute of Technology and Science , Karimnagar , Andhra Pradesh

Assistant Professor, Computer Science and Engineering Department

ABSTRACT: *Some different anonymization techniques, such as generalization and bucketization, have been designed for privacy preserving micro data publishing. Recent work has shown that 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 this paper, we present a novel technique called slicing, which partitions the data both horizontally and vertically. We show that slicing preserves better data utility than generalization and can be used for membership disclosure protection. Another important advantage of slicing is that it can handle high-dimensional data.*

*We show how slicing can be used for attribute disclosure protection and develop an efficient algorithm (**An algorithm is a procedure or formula for solving a problem.**) for computing the sliced data that obey the diversity requirement.*

*We show how slicing can be used for attribute disclosure (**uncover**) protection and develop an efficient algorithm for computing the sliced data that obey the '-diversity requirement. Our workload experiments confirm that slicing preserves better utility than generalization and is more effective than bucketization in workloads involving the sensitive attribute. Our experiments also demonstrate that slicing can be used to prevent membership disclosure. Slicing gives us a higher security as well as open source environment. i.e. on integration of project.*

Keywords :—*Privacy preservation, data anonymization, data publishing, data security*

I. INTRODUCTION

Privacy Preserving 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 [17] for 'ℓ-diversity [25].

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 Zipcode;

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

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. In bucketization, one separates the SAs from the QIs by randomly permuting the SA values in each bucket. It has been shown [1], [16], that generalization for k anonymity losses considerable amount of information, especially for high-dimensional data. This is due to the following three reasons. 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. 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. 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 [26], [17] 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.

II. Privacy Requirements

Several types of information disclosure in microdata publishing have been identified in the literature [6, 16]. An important type of information disclosure is *attribute disclosure*. Attribute disclosure occurs when a sensitive attribute value is associated with an individual. This is different from both *identity disclosure* (i.e., linking an individual to a record in the database) and membership *disclosure* [7, 23] (i.e., learning whether an individual is included in the database). As in [5], this paper considers *attribute disclosure*. *k*-Anonymity [25 , 26] (requiring each equivalence class contains at least *k* records) aims at preventing identity disclosure. Because identity disclosure leads to attribute disclosure (once the record is identified, its sensitive value is immediately revealed), *k*-anonymity can partly prevent attribute disclosure. But because attribute disclosure can occur without identity disclosure [21, 29] (for example, when all records in the equivalence class have the same sensitive value) , *k*-anonymity does not prevent attribute disclosure. Diversity [21] remedies the above limitations of *k*-anonymity by requiring that in any equivalence class, each sensitive value can occur with a frequency of at most 1/_. While there are several other definitions of _-diversity such as recursive (*c*, _) –diversity , the above probabilistic interpretation is the most widely used one in the literature.

A similar privacy requirement is the (*α, k*) - anonymity [29]. _-Diversity ensures that the probability of inferring the sensitive value is bounded by 1/_. However, this confidence bound may be too strong for some sensitive values (e.g., a common form of disease) and too weak for some other sensitive values (e.g., a rare form of cancer). *t*-Closeness [19] remedies the limitations of _-diversity , by requiring the sensitive attribute distribution in each equivalence class to be close to that in the overall data. A closely-related privacy requirement is the template - based privacy [27] where the probability of each sensitive value is bounded separately. Similar to *t*-closeness, semantic privacy [5] also tries to bound the difference between the baseline belief (i.e., the distribution in the overall population) and the posterior belief (i.e., the distribution in each equivalence class). Unlike *t*-closeness that uses Earth Mover's Distance (EMD) (which is an *additive* measure), semantic privacy uses a *multiplicative* measure which bounds the ratio of the probability of each sensitive value in each equivalence class and that in the overall distribution. One advantage of semantic privacy is that it gives a bound on the adversary's knowledge gain: classification accuracy is bounded when semantic privacy is satisfied. Semantic privacy is quite strong and it does not capture semantic meanings of sensitive values as EMD.

III. Utility Measures

It is important that the anonymized data can be used for data analysis or data mining tasks. Otherwise, one can simply remove all quasi-identifiers and output the trivially-anonymized data, which provides maximum privacy. Also, it is unclear what kinds of data mining tasks will be performed on the anonymized data. Otherwise, instead of publishing the anonymized data, one can simply perform the data mining tasks and output their results. Because of this, most utility measures are workload-independent, i.e., they do not consider any particular data mining workload. For example, the utility of the anonymized data has been measured by the number of generalization steps, the average size of the equivalence classes [21] , the discernibility metric (DM) [4] which sums up the squares of equivalence class sizes, and the KL-divergence between the reconstructed distribution and the true distribution for all possible quasi-identifier values [13]. Several researchers have proposed to evaluate the utility of the anonymized data in terms of data mining workloads , such as classification and aggregate query answering (A comprehensive discussion on the privacy-preserving data publishing is given in [9]). Classification accuracy on the anonymized data has been evaluated in [18, 28, 10, 27, 5]. The main results from these studies are : (1) anonymization algorithms can be tailored to optimize the performance of specific data mining workloads and (2) utility from classification is bounded when attributed disclosure is prevented. Aggregate query answering has also been used for evaluating data utility [30, 14, 24].

IV. Proposed Method

In this paper, we present a novel technique called slicing for privacy-preserving data publishing. Our contributions include the following. 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 ℓ- diverse slicing, which ensures that the adversary cannot learn the sensitive value of any individual with a probability greater than 1/ℓ, 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 lessfrequent 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 k^C 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. We also evaluated the performance of slicing in anonymizing the Netflix Prize data set.

V. Proposed techniques used

In the proposed work we have used slicing technique and compared it to generalization and bucketization

V.1 Slicing : Slicing first partitions attributes into columns. Each column contains a subset of attributes. This vertically partitions the table. For example, the sliced table in Table 6 contains two columns: the first column contains { Age; Sex} and the second column contains {Zipcode; Disease}. The sliced table shown in Table 5 contains four columns, where each column contains exactly one attribute. Slicing also partition tuples into buckets. Each bucket contains a subset of tuples. This horizontally partitions the table. For example, both sliced tables in Tables 5 and 6 contain two buckets, each containing four tuples. Within each bucket, values in each column are randomly permutated to break the linking between different columns. For example, in the first bucket of the sliced table shown in Table 6, the values {(22,M) , (22,F) (33,F) , (52,F)} are randomly permutated and the values{(47906, dyspepsia),(47906,flu) , (47905, flu), (47905, bronchitis)} are randomly permutated so that the linking between the two columns within one bucket is hidden.

V.1.1 Results:

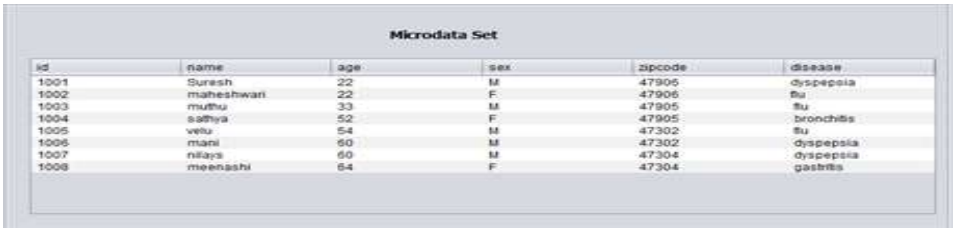

Fig 1 Original Microdata Table

Fig 2 The Generalized Table

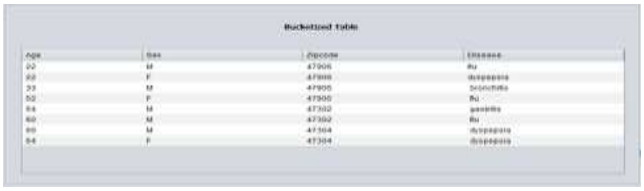

Fig 3 The Bucketized Table

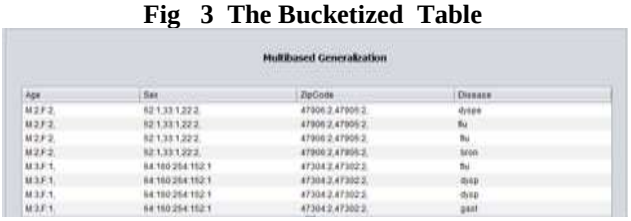

Fig 4 Multiset based generalization

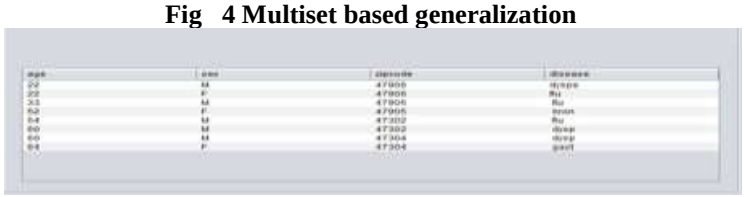

Fig 5 One attribute per column slicing

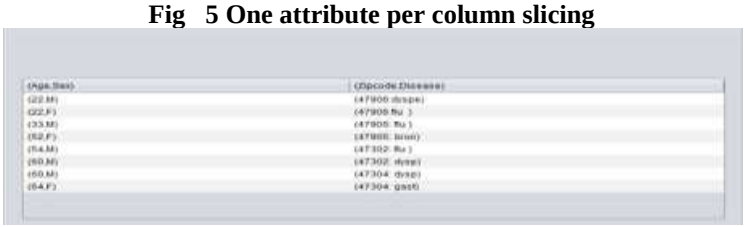

Fig 6 The sliced Table

VI. COMPARITATIVE RESULTS

Two popular anonymization techniques are generalization and bucketization. Generalization replaces a value with a "less-specific but semantically consistent" value. The main problems with generalization are: 1) it fails on high-dimensional data due to the curse of dimensionality and it causes too much information loss due to the uniform-distribution assumption .Bucketization 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 anonymized data consist of a set of buckets with permuted sensitive attribute values. In particular, bucketization has been used for anonymizing high-dimensional data [. However, their approach assumes a clear separation between QIs and SAs. In addition, because the exact values of all QIs are released, membership information is disclosed.

VII. MEMBERSHIP DISCLOSURE PROTECTION

In this section, we analyze how slicing can provide membership disclosure protection. Bucketization. Let us first examine how an adversary can infer membership information from bucketization. Because bucketization releases each tuple's combination of QI values in their original form and most individuals can be uniquely identified using the QI values, the adversary can determine the membership of an individual in the original data by examining whether the individual's combination of QI values occurs in the released data. Slicing. Slicing offers protection against membership disclosure because QI attributes are partitioned into different columns and correlations among different columns within each bucket are broken. Consider the sliced table in Table 1f. The table has two columns. The first bucket is resulted from four tuples; we call them the originaltuples. The bucket matches altogether 42 ¼ 16 tuples, 4 original tuples and 12 that do not appear in the original table. We call these 12 tuples fake tuples. Given any tuple, if it has no matching bucket in the sliced table, then we know for sure that the tuple is not in the original table. However, even if a tuple has one or more matching bucket, one cannot tell whether the tuple is in the original table, because it could be a fake tuple. We propose two quantitative measures for the degree of membership protection offered by slicing.

The first is the fake-original ratio (FOR), which is defined as the number of fake tuples divided by the number of original tuples. Intuitively, the larger the FOR, the more membership protection is provided. A sliced bucket of size k can potentially match kc tuples, including k original tuples and kc _ k fake tuples; hence, the FOR is kc_1 _ 1. When one has chosen a minimal threshold for the FOR, one can choose k and c appropriately to satisfy the threshold. The second measure is to consider the number of matching buckets for original tuples and that for fake tuples. If they are similar enough, membership information is protected because the adversary cannot distinguish original tuples from fake tuples. Since the main focus of this paper is attribute disclosure, we do not intend to propose a comprehensive analysis for membership disclosure protection. In our experiments (Section 6), we empirically compare bucketization and slicing in terms of the number of matching buckets for tuples that are in or not in the original data. Our experimental results show that slicing introduces a large number of tuples in D and can be used to protect membership information. Generalization. By generalizing attribute values into "less-specific but semantically consistent values," generalization offers some protection against membership disclosure. It was shown in [27] that generalization alone (e.g., used with k-anonymity) may leak membership information if the target individual is the only possible match for a generalized record. The intuition is similar to our rationale of fake tuple. If a generalized tuple does not introduce fake tuples (i.e., none of the other combinations of values are reasonable), there will be only one original tuple that matches with the generalized tuple and the membership information can still be inferred. Nergiz et al. [27] defined a large background table as the set of all "possible" tuples in order to estimate the probability whether a tuple is in the data or not (_-presence). The major problem with [27] is that it can be difficult to define the background table and in some cases the data publisher may not have such a background table. Also, the protection against membership disclosure depends on the choice of the background table. Therefore, with careful anonymization, generalization can offer some level of membership disclosure protection.

VIII. RELATED WORK

Two popular anonymization techniques are generalization and bucketization. Generalization [28], [30], [29] replaces a value with a "less-specific but semantically consistent" value. Three types of encoding schemes have been proposed for generalization: global recoding, regional recoding, and local recoding. Global recoding [18] has the property that multiple occurrences of the same value are always replaced by the same generalized value. Regional record [19] is also called multidimensional recoding (the Mondrian algorithm) which partitions the domain space into nonintersect regions and data points in the same region are represented by the region they are in. Local recoding [36] does not have the above constraints and allows different occurrences of the same value to be generalized differently. The main problems with generalization are: 1) it fails on high-dimensional data due to the curse of dimensionality [1] and 2) it causes too much information loss due to the uniform-distribution assumption [34]. Bucketization [34], [26], [17] 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 anonymized data consist of a set of buckets with permuted sensitive attribute values. In particular, bucketization has been used for anonymizing high-dimensional data [12]. However, their approach assumes a clear separation between QIs and SAs. In addition, because the exact values of all QIs are released, membership information is disclosed. A detailed comparison of slicing with generalization and bucketization is in Sections 2.2 and 2.3, respectively. Slicing has some connections to marginal publication [16]; both of them release correlations among a subset of attributes. Slicing is quite different from marginal publication in a number of aspects. First, marginal publication can be viewed as a special case of slicing which does not have horizontal partitioning. Therefore, correlations among attributes in different columns are lost in marginal publication. By horizontal partitioning, attribute correlations between different columns (at the bucket level) are preserved. Marginal publication is similar to overlapping vertical partitioning, which is left as our future work . Second, the key idea of slicing is to preserve correlations between highly correlated attributes and to break correlations between uncorrelated attributes thus achieving both better utility and better privacy. Third, existing data analysis (e.g., query answering) methods can be easily used on the sliced data. Recently, several approaches have been proposed to anonymize transactional databases. Terrovitis et al. [31] proposed the km-anonymity model which requires that, for any set of m or less items, the published database contains at least k transactions containing this set of items. This model aims at protecting the database against an adversary who has knowledge of at most m items in a specific transaction. There are several problems with the kmanonymity model: 1) it cannot prevent an adversary from learning additional items because all k records may have some other items in common; 2) the adversary may know the absence of an item and can potentially identify a particular transaction; and 3) it is difficult to set an appropriate m value. He and Naughton [13] used kanonymity as the privacy model and developed a local recoding method for anonymizing transactional databases. The k-anonymity model also suffers from the first two problems above. Xu et al. [35] proposed an approach that combines k-anonymity and '-diversity but their approach considers a clear separation of the quasi identifiers and the sensitive attribute. On

the contrary, slicing can be applied without such a separation. Existing privacy measures for membership disclosure protection include differential privacy [6], [7], [9] and _- presence [27]. Differential privacy [6], [7], [9] has recently received much attention in data privacy. Most results on differential privacy are about answering statistical queries, rather than publishing microdata. A survey on these results can be found in [8]. On the other hand, _-presence [27] assumes that the published database is a sample of a large public database and the adversary has knowledge of this large database. The calculation of disclosure risk depends on the choice of this large database. Finally, on attribute disclosure protection, a number of privacy models have been proposed, including '-diversity [25], ð_; kÞ-anonymity [33], and t-closeness [21]. A few others consider the adversary's background knowledge [26], [4], [22], [24]. Wong et al. [32] considered adversaries who have knowledge of the anonymization method.

IX. Conclusions And Future Work

This paper presents a new approach called slicing to privacy preserving microdata publishing. 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 analyze 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. In [22], [24], we show that attribute correlations can be used for privacy attacks. While a number of anonymization techniques have been designed, it remains an open problem on how to use the anonymized data. In our experiments, we randomly generate the associations between column values of a bucket. This may lose data utility. Another direction is to design data mining tasks using the anonymized data [14] computed by various anonymization techniques

X. 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]. 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]. Dinur and K. Nissim, "Revealing Information while Preserving Privacy," Proc. ACM Symp. Principles of Database Systems (PODS), pp. 202-210, 2003.
[7]. C. Dwork, "Differential Privacy," Proc. Int'l Colloquium Automata, Languages and Programming (ICALP), pp. 1-12, 2006.
[8]. C. Dwork, "Differential Privacy: A Survey of Results," Proc. Fifth Int'l Conf. Theory and Applications of Models of Computation (TAMC), pp. 1-19, 2008.
[9]. 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.
[10]. J.H. Friedman, J.L. Bentley, and R.A. Finkel, "An Algorithm forFinding Best Matches in Logarithmic Expected Time," ACM Trans. Math. Software, vol. 3, no. 3, pp. 209-226, 1977.
[11]. 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.
[12]. 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.
[13]. 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.
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.
[14]. L. Kaufman and P. Rousueeuw, "Finding Groups in Data: An Introduction to Cluster Analysis," John Wiley & Sons, 1990.
[15]. 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.
[16]. 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.
[17]. 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.
[18]. K. LeFevre, D. DeWitt, and R. Ramakrishnan, "Mondrian Multidimensional k-Anonymity," Proc. Int'l Conf. Data Eng. (ICDE), p. 25, 2006.
[19]. 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.
[20]. N. Li, T. Li, and S. Venkatasubramanian, "t-Closeness: Privacy Beyond k-Anonymity and '-Diversity," Proc. IEEE 23rd Int'l Conf. Data Eng. (ICDE), pp. 106-115, 2007

[21]. 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.
[22]. 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.
[23]. 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.
A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam, "'-Diversity: Privacy Beyond k-Anonymity," Proc. Int'l Conf. Data Eng. (ICDE), p. 24, 2006.
[24]. 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
[25]. 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.
[26]. 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
[27]. 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.
[28]. [30] 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.
[29]. 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. [32] 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.