A survey on Microaggregation based Privacy Preserving Data Mining Techniques

V. Jane Varamani Sulekha[1], Dr. G. Arumugam[2]
[1](Department of Information Technology, Fatima College, Madurai
Email: sulekhaisaac@gmail.com)
[2] (Department of Computer Science, Madurai Kamaraj University, Madurai
Email: gurusamyarumugam@gmail.com)

ABSTRACT
Massive collection of personal information in big data era has its own confidentiality threats which results in leakage of sensitive data. In order to get the benefits of data mining without degrading confidentiality, it is necessary to include data protection techniques as core element of data mining. Implementing the confidentiality and data safeguarding schemes plays a vital role in data mining. When applying privacy preservation techniques, importance is given to the utility and information loss. In this paper we analyze different microaggregation data protection techniques. Microaggregation techniques try to protect individual records in such a way that data can be mined, distributed and published without providing any personal information that can be associated with specific individuals.

Keywords *- Data Mining, Microaggregation, Privacy Preservation, Privacy, PPDM*

I. INTRODUCTION

Due to the enormous advancement in storing, processing, and networking capabilities of computing devices, there has been a tremendous growth in the collection of digital information about individuals. And the development of new computing paradigms, such as cloud computing, increases the possibility of large-scale distributed data collection from multiple sources. While the collected data offer incredible opportunities for mining useful information, there is also a threat to privacy because data in raw form often contain sensitive information about individuals. Data mining or knowledge discovery from data (KDD), aims to find out interesting patterns and knowledge from big data. Although its application has been mostly successful, data mining can set up compromising situations for sensitive information, which is a serious privacy threat. The response to this problem has been significant enough to lead to Privacy preserving data mining (PPDM), the goal of which is to safeguard information from unsolicited or unauthorized disclosure while preserving the data's utility. PPDM techniques aim to avoid the direct use of sensitive raw data, such as an individual's ID and mobile numbers and attempt to ignore sensitive patterns in mining results, such as clues to unidentified personal information derived from a consumer's shopping behavior. PPDM studies how to transform raw data into a transformed version that is immunized against privacy attacks but that still supports effective data mining tasks.

Privacy is defined as "protecting individual's personal information". Protection of privacy has become an important concern in this big data era. Defining privacy is a challenging task. From literature [13], roughly 87 percent of the U.S. population can be uniquely identified based on gender, date of birth and zip code. Simply removing explicit identifiers (e.g., name) does not preserve privacy. The adversary may know these quasi identifiers from publically available sources such as a voter list. An individual person can be recognized from published data by simply joining the quasi identifiers, identifiers and sensitive attributes with an external data source.

There are many applications where PPDM can be presented to provide useful knowledge while meeting accepted standards for protecting privacy. As an example, consider mining of supermarket transaction data. Most supermarkets now offer discount cards to consumers who are willing to have their purchases tracked. Generating association rules from such data is a commonly used data mining example, leading to insight into buyer behavior that can be used to redesign store layouts, develop retailing promotions, etc. Privacy issues in Data Mining have become more important in recent years because of the development in hardware and software by which we can able to store huge volume of user data and the new and advanced data mining algorithms to act upon this information.

PPDM can be expressed as two parts. One is Anonymizing Sensitive attributes and another one is

preserving mined rules. In the first part, sensitive attributes (identifiers, quasi identifiers, sensitive attributes) such as name, age, income, disease, address, phone number, SIN (social insurance number), SSN (social security number), should be eliminated or anonymized from the original database, so that the receiver of the data do not interfere into another person's privacy. Next, sensitive rules and sensitive patterns mined from a database by using data mining algorithms should also be preserved because that too may compromise data privacy. A number of privacy-preserving data mining methods have been proposed which take either a cryptographic or a statistical approach. The cryptographic approach ensures strong privacy and accuracy via a secure multi-party computation, but typically suffers from its poor performance and extremely expensive. The statistical approach has been used to mine decision trees, association rules, and clustering, and is popular mainly because of its high performance.

The primary objective in PPDM is to develop privacy preservation techniques for transforming the original data, so that the sensitive data and sensitive rules can be preserved. PPDM was first suggested by [4] and [5]. To overcome the privacy issues, various solutions have been proposed by researchers. PPDM techniques available in the literature are as follows. Blocking - replacing the sensitive value with '?'. Perturbation - changing the value of an attribute mostly using noise, Anonymization - encrypting or removing sensitive attribute, Aggregation – aggregating the values, Swapping - interchange of values, Sampling- taking only a sample of population, Sanitization - transforming sensitive attribute, Differential privacy - maximizing the accuracy of queries from statistical databases while minimizing the chances of identifying its records. Condensation - reducing data set, Cryptography - secure transmission of data using protocols, Evolutionary algorithms like Genetic Algorithm, Artificial bee colony algorithm and Ant colony optimization algorithm transforms the original data. There are other techniques that emphasis on protecting the confidentiality of mining rules and patterns discovered from data.

This paper reviews microaggregation based privacy preservation techniques with its merits and demerits. Section 2 presents PPDM architecture, Section 3 describes Microaggregation, Section 4 lists different Microaggregation methods, Section 5 gives idea about evaluation criteria and Section 6 describes Conclusions and future work in the perspective of privacy preserving data mining.

II. PPDM ARCHITECTURE

Fig.1 describes the architecture for PPDM. In data mining or knowledge discovery from databases (KDD) process, the data is collected by single or various organizations and stored at respective databases. Then, it is transformed to a format suitable for analytical purposes and stored in large data warehouses. PPDM techniques are applied to this data to ensure privacy after that data mining algorithms are applied on it for the generation of information/knowledge. PPDM techniques consist of four stages. In the first stage, raw sensitive data or databases are collected from transactional databases. The second stage is applying PPDM techniques to raw sensitive data for transformation. The third stage is applying different data mining algorithms. The fourth stage is the output of different data mining algorithms and methods without any privacy breach.

- At stage 1, the raw data compiled from a single or multiple databases or even data marts are transformed into a format that is well suited for analytical purposes.
- At stage 2, PPDM technique can be applied to the data warehouse. Sensitive attribute, quasi identifiers and identifiers are removed at this stage. The techniques applied at this stage are anonymization, perturbation, blocking, suppression, microaggregation, differential privacy, modification, generalization, sampling etc.
- At stage 3, data mining algorithms are applied to the transformed data for knowledge discovery. In some cases even the data mining algorithms are modified for the purpose of protecting privacy without sacrificing the goals of data mining.
- At stage 4, different useful patterns and rules are generated. The information/knowledge so revealed by the data mining algorithms is checked for its sensitiveness towards disclosure risks.

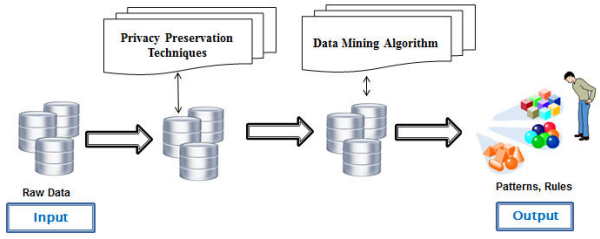

Fig. 1 PPDM Architecture

III. MICROAGGREGATION

Microaggregation is a perturbation based privacy preservation technique. In Microaggregation the individual values are replaced by values computed on

small aggregates prior to publishing. In other words, instead of releasing the original values of the individual records, the system releases the mean of the group (or median, mode, weighted average) to which the values belongs. Microaggregation technique consists of two phases, partitioning and aggregation. In partitioning, the original micro dataset is partitioned into several disjointed clusters/groups so that all records in the same group are very much related to each other and, simultaneously, dissimilar to the records in other groups. Moreover, each group is forced to have at least k records. Fig. 2 describes the microaggregation method by using mean.

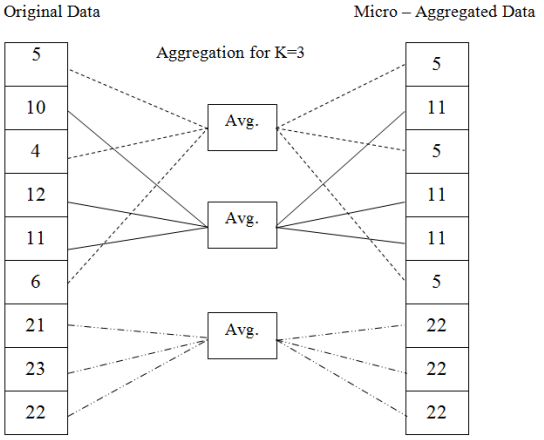

Fig. 2 Example of Microaggregation using Mean

A microdata set is a file or a table with n records and m attributes. The attributes can be classified into four classifications, generally they are not disjoint. The four types are Identifiers, Quasi-Identifiers, Confidential Outcome attributes (sensitive attributes) and Non-Confidential attributes. Identifiers are used to identify the individual person. Passport Number, Aadhaar Number, Name and social security number are examples of identifiers. A combination of Quasi-identifiers can be used to identify the individual person. Address, gender, age, telephone and pincode are examples of Quasi-Identifiers. Sensitive attributes describe the individual person. Religion, Health Condition and Salary are few examples of sensitive attributes. Non confidential attributes will not reveal any sensitive information about the person. A microdata set with n records can be micro aggregated by forming different groups with size at least k. Every attribute is substituted with the average of the group that the attribute belongs. Usually groups are formed with highest similarities. After updating the original value, the resulting records can be released for mining. The ideal k-partition with minimum information loss is defined to be the one that maximizes the group similarity. The higher the group similarity, the lower the information loss.

Microaggregation replaces values in a group by the group mean. The sum of squares condition is used to measure the similarity in clusters. Within the group sum of squares SSE is specified as

$$SSE = \sum_{i-1}^{g} \sum_{j=1}^{n_i} (x_{ij} - \bar{x}_i)' (x_{ij} - \bar{x}_i) \quad (1)$$

The similarity is higher when SSE is lesser in the cluster. The between group sum of squares SSA is specified as

$$SSA = \sum_{i=1}^{g} n_i (\bar{x}_i - \bar{x})' (\bar{x}_i - \bar{x}) \quad (2)$$

The total sum of squares SST is specified as

$$SST = \sum_{i-1}^{g} \sum_{j=1}^{n_i} (x_{ij} - \bar{x})' (x_{ij} - \bar{x}) \quad (3)$$

Information Loss (IL) is standardized between 0 to 1 and defined as

$$IL = \frac{SSE}{SST}$$

3.1 Evolution of Microaggregation

The concept of microdata based microaggregation was first introduced by Defays, Nanopoulos [10]. In microaggregation method Individual data in the whole data set is replaced by averages of small aggregates.

3.1.1 Fixed Size & Variable Size Microaggregation

To handle large volume of data, the whole data set is partitioned into fixed size groups/clusters. After grouping the whole data set, each group is replaced with its group aggregates. This method is known as fixed size microaggregation. To find the optimal partitioning among all fixed size partitioning, Hanani proposed multicriteria dynamic clustering method. Here the cluster size is not a fixed one. This method is known as variable size partitioning. Depending on the group size, microaggregation is divided into fixed size microaggregation and variable sixe microaggregation. Fixed Size microaggregation depends on the the total number of records *n* and the anonymity parameter *k.* In fixed size microaggregation, dataset is partitioned into fixed sized groups with k values but one cluster which has a size between *k* and *(2k−1).* In variable size microaggregation groups have variable sizes. Fixed size microaggregation takes less computation time in partitioning the dataset, but the variable size partition method is more flexible.

3.1.2 Univariate & Multivariate Microaggregation

Unfortunately there is no guarantee that this method will always reach the optimum. Another problem is the quality of the micro-aggregated data. It is easy to see that if the variables are not all correlated, the groups will be heterogeneous and the method will transform drastically some of the variables. In order to avoid the loss of information caused by aggregation in clusters of fixed size of the original units, it has been proposed that the different unidimensional variables (univariate variable) be aggregated separately, by ranking the values assumed by these variables and by an aggregation in fixed size groups of contiguous values. Univariate method underlines the necessarily separate treatment of the different individual variables which results in separate classifications and aggregations of units.

Univariate microaggregation refers to any of the following two cases:

1) Microdata set being micro aggregated consist of single variable.
2) Microdata set being micro aggregated consists of several variables, but those are considered independently in turn. Data vectors are ranked by the first variable and the values of the variable are micro aggregated. Then the same procedure is repeated for the second variable and so on.

This univariate variable idea leads to concept of Multivariate variable. A set of p variables can be treated as a single multivariate variable, resulting in a single grouping, or as separate p variables, resulting in p groupings each. Multivariate microaggregation refers to techniques which allow simultaneous microaggregation of several variables so that a single k-partition for the whole data set is obtained.

Complexity is lesser when single variable is involved, at the same time utility should be considered. In Multivariate microaggregation, the grouping process is applied to sets of variables of the microdata set. In this case, when all the variables are micro aggregated together, k-Anonymity is automatically satisfied thereby reducing the risk of data disclosure. Thus, one can concentrate in maximizing data utility. More generally, the initial vector of variables can be segmented into a number of variables, multivariate or univariate, which can be called as segments. Each segment is treated separately. Figure 3, describes the evolution of microaggregation methods.

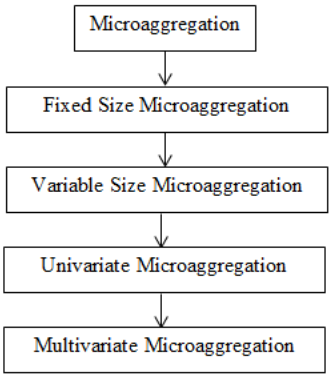

Fig. 3 Evolution of Microaggreation Methods

There is another concept data oriented microaggregation [15] is also available in the literature. In the data oriented microaggregation, the partition is based on the data with group sizes between k and (2k-1).

IV. DIFFERENT MICROAGGREGATION METHODS

This section gives a brief overview of the various approaches to microaggregation method which exists in the literature.

4.1 Practical Data Oriented Microaggregation (PDOM)

Practical Data oriented microaggregation [5], depends on the cluster size k and total numbers of records n. Each cluster consists of k values. There may be one cluster with a size between k and (2k-1). It. Univariate microaggregation based on Wards hierarchical clustering, Univariate microaggregation based on Genetic algorithm, Multivariate fixed size microaggregation and Multivariate microaggregation based on Wards hierarchical clustering are discussed. If the data set is very large or if microaggregation is to be done on-line then genetic method is good at speed and minimum information loss. When considering data disclosure, K - Ward based fixed size microaggregation is safe. Both univariate and multivariate microaggregation are addressed here.

4.2 Maximum Distance based Microaggregation

The Maximum Distance (MD) Method [5] is proposed with univariate and multivariate microaggregation method. The advantage of this method is its performance and simplicity. The algorithm builds a k-partition as follows. Two distant records r, s are identified using Euclidean distance. Subsequently two groups are formed with the first group with r and the (k-1) records closest to r and second group with s and the (k-1) records closest to

s. If there exist any records, which do not belong to any of the groups, implement the same strategy iteratively to form new groups. Finally, we will obtain a k-partition of the data set. After the partition, each record in the group is replaced with its centroid.

4.3 Optimal microaggregation

Computational complexity of optimal microaggregation [6] with minimal information loss for a fixed security level is proposed. They have shown that the problem of optimal microaggregation is NP-hard. This provides theoretical justification for the use of heuristic approaches in microaggregation.

4.4 Shortest path algorithm based microaggregation

A polynomial algorithm for microaggregation [7], expresses the microaggregation problem as a shortest path problem on a graph. First graph is constructed. Then each arc of the graph corresponds to a possible group may considered as part of an optimal partition. Each arc is labeled by the error that would result if that group were to be included in the partition. This algorithm takes advantage of the following two cases. 1. Under an optimal partitioning the observations in each group are contiguous if the observations are sorted in an ascending order. 2. In every optimal partition, each group has between k and 2k-1 observations.

4.5 Maximum Distance to Average Vector Method (MDAV)

Maximum Distance to Average Vector Method (MDAV) [8], is a microaggregation method proposed in the µ-Argus package for statistical disclosure control. It is based on establishing groups based on the distance between distinct data and the centroid. First, a square matrix is computed which consists of distances between all records. Two main methods can be adopted to perform these distance calculations. In the first method, distances are calculated and stored in the beginning of the microaggregation process. In the second method, distances are calculated dynamically. The first method is computationally inexpensive but it requires too much memory space when the number of records in the data set is large. Subsequently MDAV form two groups. To build these groups, the centroid c, is calculated. Then the most distant record ‘r’ from ‘c’ is taken and a group of ‘k’ records is built around ‘r’. The group of ‘k’ records around ‘r’ is formed by ‘r’ and the (k-1) closest records to ‘r’. Next, the most distant record,‘s’ from ‘r’ is taken and a group of ‘k’ records is built around ‘s’. The formation of groups continues until the number of remaining records (NORR) is less than 2k. When this condition is met, two cases are possible, namely NORR < k or NORR ≥ k. In the first case, the remaining records are assigned to their closest group. In the second case, a new group is built with all the remaining records.

Algorithm: MDAV

1. *Calculate centroid c of dataset D.*
2. *Find the most distant record r from the centroid c.*
3. *Build group g_i with (k-1) closest records to r.*
4. *Find the most distant record, s from r.*
5. *Build group $g_{(i+1)}$ with (k-1) closest records to r.*
6. *Repeat the steps 1 to 5 till there are more than (2k-1) records left to be assigned to any group.*
7. *If there remains more than (k-1) records to be assigned then form a new group with the remaining records.*
8. *Assign the remaining records to the closest group.*
9. *Build a micro aggregated data set D' by replacing the records with its mean value of the group to which it belongs.sdsds*

4.6 Minimum Spanning Tree Partitioning (MSTP)

Minimum Spanning Tree Partitioning (MSTP) for microaggregation [9] is a variable size multivariate microaggregation method. Minimum Spanning Tree (MST) is built using Prim Method. Standard MST partitioning algorithm does not consider the group size, so that it cannot be used in microaggregation problem. To overcome this problem, a small alteration is made in the MSTP algorithm. That is oversized clusters are further divided into small clusters.

Algorithm: MSTP

1. MST construction: Create the minimum spanning tree over the data points using Prim’s algorithm.
2. Edge cutting: Iteratively visit every MST edge in length order, from longest to shortest, and delete the removable edges (when cut, resulting clusters do not violate the minimum size constraint), while retaining the remaining edges. This phase produces a forest of irreducible trees (tree with all non-removable edges) each of which corresponds to a cluster.
3. Cluster creation: Traverse the resulting forest to assign each data point to a cluster.
4. Further dividing oversized clusters: Either by the diameter-based or by the centroid-based fixed size method

4.7 Multivariate Data Oriented microaggregation (MDOM)

Hansen–Mukherjee's algorithm for optimal univariate microaggregation is used to enhance the existing heuristics for multivariate data oriented microaggregation [10]. The basic idea is to use fixed size heuristics or other algorithms such as nearest point next (NPN) to construct a path navigating all points in a multivariate dataset. After that the multivariate variation of Hansen–Mukherjee's algorithm (MHM) is used on that path. The result is a data oriented k-partition. The NPN selects the first record by computing the record utmost away from the centroid of the entire dataset. The record closest to the first record is selected as the second record. The third record is closest to the second record. This process continues until all of the records have been added to the tour. MHM constructs a graph based on an ordered list of records, and finding the shortest path in the graph. The arcs in the shortest path correspond to a partition of the records that is guaranteed to be the lowest cost partition consistent with the specified ordering. Four heuristics methods NPN-MHM (Nearest Point Next - Hansen–Mukherjee algorithm), MD-MHM (Maximum Distance - Hansen–Mukherjee algorithm), MDAV-MHM (Maximum Distance to Average Vector - Hansen–Mukherjee algorithm), CBFS-MHM (Centroid Based Fixed Size - Hansen–Mukherjee algorithm) are also addressed here.

4.8 Variable-MDAV 2006

V-MDAV [11], implements a similar approach of MDAV. Distances between any two records are built as a Matrix. Then the global centroid of the whole data set is calculated and the most distant record from this global centroid is searched. Once the distant record is found, a cluster of k records is made by selecting the (k-1) records closest to the initial one. MDAV method iteratively applies this technique, until all records in the data set are assigned to a cluster. But V-MDAV applies data set distribution technique and produces variable-size groups. Expanding the group is determined by the following formula, Unassigned record < γ (Shortest distance from the unassigned record to another unassigned record). Gain factor γ has to be tuned according to the data set. V-MDAV is equal to MDAV when γ= 0. On the contrary, when the data set is clustered the best values for γ are usually close to one. The authors chosen γ= 0.2 for dispersed data sets and γ= 1.1 for grouped data sets. MDAV calculates a centroid in each iteration, but V-MDAV calculates the data set centroid act the beginning.

Algorithm: V-MDAV

1. *Calculate distance matrix of the dataset D.*
2. *Compute centroid C of dataset D.*
3. *Select the most distant record x from the centroid C.*
4. *Build group g_i with (k-1) closest records to x.*
5. *Extend the groupg_i.*
6. *Repeat the steps 3 to 5 till there are (k-1) records left to be assigned to any group.*
7. *Assign the remaining unassigned records to its closest group.*
8. *Build a micro aggregated data set D^' by replacing the records with its mean value of a group to which it belongs.*

4.9 Genetic-Algorithm-Based Microaggregation (2006)

A new method for multivariate microaggregation [12] which is based on genetic algorithms is proposed. New coding and initialization schemes have been presented. The influence of the main parameters of the GA - mutation rate, crossover rate and population size, has been experimentally and statistically examined.

A GA is a method for moving from one population of chromosomes to a new population by using a form of natural selection together with the genetics stimulated operators of crossover, mutation, and inversion. Binary coding was employed to resolve the problem of univariate microaggregation. The approximation was to sort univariate records in ascending order and to represent the i^{th} record by the i^{th} symbol in a binary string (chromosome). Therefore, a chromosome represents a k-partition as follows. If a cluster starts at the ith record, its number in the chromosome is initialized with a "1" symbol, otherwise it is set to "0". This coding makes sense for univariate data sets because they can be sorted but it cannot be suited for multivariate data set. To overcome this limitation, a new coding technique which can be applied to data sets containing any number of dimensions is introduced here. Initializing the population approach in GA, initialize the population of chromosomes through a uniform pseudo random generator mapped on the desired alphabet. After this step, we can get a partition built up by groups of cardinality at most (2k-1). However, there can be groups with less than k records. Thus, after applying initialization algorithm, we can arbitrarily move records from groups with cardinality more than k to groups under cardinality k. Finally we can get a k-partition which is candidate to be optimal. Throughout the evolution process, some chromosomes may not be candidate optimal k-partitions due to the effect of

genetic operators. The fitness function approach in GA, do not remove the chromosome from the population; instead, penalize it in order to nearly prevent its reproduction. By disciplining these undesirable chromosomes, we can reduce the search space and avoid subsequent computation of their SSE.

4.10 Two Fixed Reference Points based Microaggregation

Two Fixed Reference Points (TFRP) based microaggregation [13] is proposed. TFRP has two steps and its two steps are denoted as TFRP I and TFRP II. In the first phase, fixed size algorithm is used to partition the data set. In the second phase, it reduces the number of partitions produced by the first phase to improve the data quality.

In the first phase, TFRP uses a unique fixed size algorithm to select two fixed reference points, *R*1 and *R*2, which are two extreme points calculated from the micro data set. From these two reference points, TFRP selects any one reference point. In each iteration, TFRP selects the *initial point* x_r, the furthest vector from a reference point, and then computes the distance of each vector to x_r. After this, TERP selects the *k*-1 closest vectors to x_r and x_r itself to form group G_r. After *n* iterations, each remaining vector is added to its nearest group.

Algorithm 1: TFRP I

1. *Calculate the two reference points R1 and R2. All vectors are assigned to a set (SET).*
2. *Select a reference point.*
3. *Select an initial point xi from the reference point.*
4. *Calculate the distance of each vector to xi.*
5. *Select k-1 closest vectors together with xi to form a group, and remove the k vectors from SET.*
6. *Select another reference point, then go to Step 2 until* $|SET| < k$.
7. *Assign each remaining vector of SET to its closest group.*

In the second phase, within-group squared error (GSE) of a Group, total Sum of the within-group squared errors (SSE) and the total sum of square errors (SST) are calculated. TFRP sorts groups in decreasing order according to their GSE values. Then, TFRP selects the groups in order and to check whether changing the members of the selected group to their nearest groups can increase the data quality.

Algorithm 2: TFRP II

1. Calculate GSE of each group, and sort them in decreasing order.
2. Select a group G_i in order and calculate the current total sum of the within-group squared errors (*SSE*1).
3. Calculate the distance of each vector of G_i to any other group.
4. Allocate each vector of G_i to its closest group provisionally, and compute the current total sum of the within-group squared errors (*SSE*2).
5. If *SSE*1 > *SSE*2, then allocate each vector of G_i to its closest group; otherwise, regain G_i.
6. Return to Step 2 and repeat until each group is checked.

4.11 Microaggregation based heuristics for p-sensitive k-anonumity

Evolution of k-anonymity called p-sensitive k-anonymity [14] is presented. Its idea is that there are at least p different values for each sensitive attribute within the records sharing a combination of key attributes. The algorithm receives micro data set X consisting of n records having Q numerical key attributes and L discrete confidential attributes each as input. The result of the algorithm is a k-partition used to micro aggregate the original micro data set and to generate a micro aggregated data set X' that fulfills the p-sensitive k-anonymity property.

The algorithm constructs the initial clusters that fulfill the p-sensitive k-anonymity property. They adopt two selection methods to build a cluster. The first selection method is based on Maximum Distance to Average Vector (MDAV), it selects the average vector of the records that remain unassigned and select the record which is furthest from the average. Next method, p-Sensitive k-anonymity with Random Seeds selects initial records which is mainly random. The previous selection method fails to correctly allot the clusters amongst the complete data set. Thus, the information loss in terms of SSE grows. In order to overcome this limitation, the second method is proposed.

Algorithm 1 *p-Sensitive k-anonymity micro-aggregation-based heuristic*

$x_1; x_2;$: : : ; x_n: the records in the original data set X. The set of confidential attributes. Q: the set of key attributes. x_j(Q): the projection of record x_j on its key attributes. k: the minimum number of records per group. p: the minimum number of different values for each

confidential attribute in a group. P: an initially empty partition. UR: the set of records X which have not been assigned to any group yet.

1. i := 0
2. while (Cardinality(UR) >= k and UR contains at least p different values for each attribute in L) do
3. x_r := SelectRecordToBuildCluster();
4. C_i := newEmptyGroup();
5. C_i := AssignRecordToGroup(C_i, x_r);
6. while (confidential attributes of the records in C do not satisfy p-sensitivity) do
7. Take x_s ϵ UR so that x_s (Q) is the nearest record to x_r (Q) that contributes to the compliance of p-sensitivity
8. C_i := AssignRecordToGroup(C_i, x_s);
9. end while
10. while (Cardinality(C_i) < k) do
11. x_s := ElementWithMinimumDistance(UR, x_r));
12. C_i := AssignRecordToGroup(Ci, x_s);
13. end while
14. P := AddGroupToPartition(C_i,P);
15. i := i + 1
16. end while
17. for (for all x ε UR) do
18. i:= ClosestGroup(x, P);
19. C_i := AssignRecordToGroup(C_i, x);
20. end for
21. for (j = 1 to n) do
22. x_j' := x_j with x_j (Q) replaced by Centroid(C(Q)), where C is the group in P to which x_j has been assigned.
23. end for
24. Return the micro-aggregated, p-sensitive, k-anonymous data set .

4.12 Hybrid microdata using microaggregation

A new method called microaggregation based hybrid data [15] achieved by mixing the original data and synthetic data have been proposed. It combines the strengths of masked data having flexible utility and synthetic data with low disclosure risk.

This technique has two steps:

1)A generic synthetic data generator, it generates a synthetic data set.
2)A microaggregation heuristic, which generates micro aggregated data.

One can adopt any synthetic data generator in this approach. Microaggregation stage consists of three steps. In the first step, clusters containing between k and 2k-1 records are generated. In the second step, a synthetic version of each cluster is generated. Finally in third step, the original records in each cluster are replaced by the records in the equivalent synthetic cluster

1. Group the dataset into clusters containing k and (2k-1) records.
2. Apply a synthetic data generator algorithm to achieve a synthetic version of each cluster.
3. Substitute the original records in each cluster by the records in the equal synthetic cluster

One of the following two microaggregation method can be used to replace the original value.

1. MDAV generic with the arithmetic mean as average operator
2. Variable size MDAV based microaggregation.

The micro hybrid method is a simple approach to preserve privacy of data. It can be applied to any data type and can yield groups of variable size.

4.13 Density based Microaggregation

A Density Based Algorithm (DBA) based microaggregation [16] is proposed. The DBA has two phases. First Phase, partitions the data set into groups in which each group has at least k records. To partition the data set, it uses K nearest neighborhood of the record with the maximum k-density among all the records that are not allocated to any group. The grouping procedure continues till k records remain unallocated. These remaining k records are then allocated to its nearest groups. The second phase, further tune the partition in order to achieve small information loss and maximum data utility. Second phase may decompose the formed groups or may merge its records to other groups.

4.14 Median based Microaggregation

Microdata Protection Method through Microaggregation based on Median [17] is proposed. It divides the entire microdata set into a number of mutually exclusive and wide-ranging groups. After this grouping, it publishes the median of each group instead of individual records. It guarantees that the modification has no effect and the modified data and the original data are similar in this method. As micro aggregated data causes information loss, it uses sum of absolute deviations from median (ADM) as a measure of distortion that is always less than the distortion measure sum of squares of errors (SSE).

4.15 Data Recipient centered Microaggregation

A data recipient centered de-identification method to retain statistical attributes [18] is proposed. Based on the input from the receiver (the researcher) de-identification can be done because the researchers have a plan of how to use the data. In this approach, modified version of the condensation technique is adopted. Condensation clusters similar records into groups just like microaggregation technique. However, instead of masking only the values of quasi identifier attributes in the groups, condensation replaces the values of all attributes with synthetic data that was randomly generated based on the statistical attributes of the original data. Linear regression, Logistic regression and Cox's proportional hazards model can be used to model the synthetic data set.

The method then runs the k-means clustering algorithm to generate clusters. The objective of k-means clustering is to minimize the average squared Euclidean distance of data points from their cluster centers, where a cluster center is defined as the mean or centroid of the data points in a cluster. Weighted Euclidean distance function is used to calculate the distances among the data points. To ensure privacy, original data are replaced with synthetic data. Synthetic data is generated for all attributes in each cluster using the computed mean and covariance statistics.

The widespread approach of this method is as follows:

1. Ask for input from the data recipients about their data usage plans (mining methods, statistical analyses, etc.).
2. Analyze the proposed data mining and statistical models.
3. Design a de-identification method that will minimally obscure the data while ensuring privacy.

4.16 T-Closeness through Microaggregation

T-Closeness through Microaggregation [19] is proposed. The advantage of microaggregation is also presented. Three microaggregation-based algorithms for t-closeness are addressed.

4.16.1 t-Closeness through microaggregation and merging of microaggregated groups of records.

This algorithm consists of two stages. First microaggregation takes place and then the algorithm merges clusters until the t-closeness is satisfied. In the microaggregation step any regular microaggregation algorithm can be used because the implementation of t-closeness takes place only after microaggregation. As a result, the algorithm is pretty clear, but the utility of the anonymized data set may be far from optimum level. If, instead of accepting the enforcement of t-closeness to the second step, we make the microaggregation algorithm aware of the t-closeness constraints at the time of cluster formation.

4.16.2 K-Anonymity-first t-closeness aware microaggregation algorithm.

This algorithm first produces a cluster of size k based on the quasi identifier attributes. After that the cluster is iteratively refined until t-closeness is satisfied. In the enhancement, the algorithm checks whether t-closeness is satisfied and, if it is not, it selects the closest record not in the cluster based on the quasi-identifiers and swaps it with a record in the cluster selected so that the Earth Mover's distance (EMD) to the distribution of the entire data set is minimized. EMD calculation takes time so the algorithm is little bit slowly in performance.

4.16.3 t-CLOSENESS AWARE MICROAGGREGATION: t-CLOSENESS-FIRST

T is a data set with n records. Group the records in T into k subsets based on the confidential attribute and then generate clusters based on the quasi-identifiers with the constraint that each cluster should contain one record from each of the k subsets (the specific record is selected based on the quasi identifier attributes). Group the records into k sets with [n/k] records, then r = n mod k records remain. Assign the remaining r records to one of the subsets. Then, generate the clusters, two records from this subset are added to the first r clusters. To minimize the influence over the EMD, we need to reduce the work required to assign the probability mass of the extra record across the whole range of values. Hence, the extra record must be close to the median record of the data set. Finally we get k-anonymous t-close data set.

4.17 Individual Ranking based Microaggregation

In order to reduce the amount of noise needed to satisfy differential privacy, Utility Preserving Differentially Private Data Releases via Individual Ranking Microaggregation [20] is proposed. This method builds on microaggregation based anonymization, which is more flexible and utility preserving than alternative anonymization methods used in the literature. By using this technique, we can improve the utility of differentially private data releases. This can be possible by Individual Ranking. In individual ranking, each variable is treated independently. Data vectors are sorted by the first variable, then groups of k successive values of the first variable are formed and, inside each group,

values are replaced by the group centroid. A similar procedure is repeated for the rest of variables. Microaggregation is done for each variable in turn so that a different partition is obtained for each variable in the microdata set.

Algorithm: *Individual Ranking based Microaggregation*

1. Use individual-ranking microaggregation independently on each attribute A_i, for i = 1 to m. Within each cluster, all attribute values are substituted by the cluster centroid value, so each microaggregated cluster consists of k repeated centroid values. Let the resulting microaggregated data set be X^M.
2. Add Laplace noise independently to each attribute A_i^M of X^M, where the scale parameter for attribute A_i^M is $\Delta(A_i^M)/\varepsilon = \Delta(A_i)/(k \text{ X } \varepsilon)$. The same noise perturbation is used on all repeated centroid values within each cluster.

V. EVALUATION CRITERIA

The following are some useful metrics which will be useful to select the best appropriate privacy preserving techniques for the data, with respect to some specific parameters.

1. **Performance (Computational time):** time required to achieve the privacy criteria.
2. **Data Utility (Information Loss):** after applying the microaggregation technique, data set should ensure minimum information loss or minimum loss in the functionality of the data.
3. **Uncertainty level:** It is a measure of uncertainty with which the sensitive information that has been microaggregated can still be reconstructed.
4. **Resistance:** Resistance is a measure of tolerance. After applying microaggregation techniques the data set can be used with various data mining algorithms and models.
5. **Disclosure Risk:** Deriving Sensitive and Private Information from the original dataset.
   i. **Attribute disclosure.** Attribute disclosure takes place when an attribute of an individual can be defined more accurately with access to the released statistic than it is possible without access to that statistic.
   ii. **Identity disclosure.** Identity disclosure takes place when a record in the protected dataset can be connected with a respondent's identity. Two main approaches are commonly applied for measuring identity disclosure risk: uniqueness and re-identification.

VI. CONCLUSION

We have summarized various Microaggregation based privacy preserving data mining techniques, and analyzed their merits and demerits. Table 1 gives various merits and demerits. There won't be any single techniques, which satisfy performance, utility, cost, complexity and tolerance. One technique may perform better than another on one particular measure. Microaggregation techniques applied to data, may consider the factors such as Privacy loss, Information loss, Data mining task, Data dimension and Volume, Data Type, Resistant to various data mining algorithms, Complexity and cost.

We have found that noise addition based microaggregation may perofrm well in numerical data set. So our further research will be on the direction of noise addtion based microaggregatin.

Table 1. Microaggregation Techniques with Merits and Demerits

Sr. No.	*Techniques*	*Merits*	*Demerits*
1	Practical Data oriented Microaggregation (1998)	• Univariate Fixed Size method reduces space complexity. • Univariate microaggregation preserves more information. • Minimum information loss in Multivariate microaggregation.	Disclosure risk is higher in Univariate microaggregation.
2	MD (1998)	Effective Portioning is possible.	Computational complexity is higher
3	Optimal microaggregation (2001)	Minimal Information Loss	-
4	SP (2003)	Minimal Information Loss	Based on multivariate data should be researched further
5	MDAV (2005)	• Computational complexity is better than MD. • Performance is good.	Not Flexible
6	MST (2005)	Better Performance	not efficient enough for

			massive data sets
7	*MDOM* (2006)	• Minimal Computing time. • Minimal Information Loss.	Fixed size microaggregation is effective than Variable size microaggregation.
8	V-MDAV (2006)	• Faster • Minimal Computing Time	optimal value of γ is studied further
9	Genetic-Algorithm-Based Microaggregation (2006)	• Better Performance than MDAV	Canonical mutation and crossover operators only tested. Modification of the genetic operators should be researched further.
10	TFRP (2007)	• Low information loss • Minimal running time • Faster	Maximum Time complexity
11	Micro-aggregation-based heuristics for p-sensitive k-anonymity (2008)	• Minimal Information Loss. • shortcomings related to generalization and suppressions are eliminated	Can combine other anonymity technique for further research.
12	Hybrid microdata using microaggregation (2010)	• Data utility • Lower disclosure risk	May have a chance of information loss
13	Density based microaggregation (2010)	• Minimal Information Loss. • Works good with univariate numerical value.	Multivariate Categorical and mixed data values should be investigated further.
14	Median based Microaggregation (2011)	• Minimal information loss	applicable for only numeric attributes
15	Data Recipient centered Microaggregation (2014)	• Utility based method • Better performance	Time complexity is not addressed.
16	T-Closeness through Microaggregation (2015)	• Minimal Computing time. • Data utility	Adaptation to categorical data can be researched further.
17	Individual ranking based Microaggregation (2016)	• Scalable • Minimal Computing time. • Minimal information loss	Other type of noise addition can be researched further.

REFERENCES

[1] Chang, C. C., Li, Y. C. and Huang, W. H. (2007) "TFRP : An efficient microaggregation algorithm for statistical disclosure control", Journal of Systems and Software,Vol. 80, No. 11, pp. 1866–1878.

[2] Defays, D. and Nanopoulos, P. (1993). «Panels of enterprises and confidentiality: the small aggregates method», in Proceedings of the 1992 Symposium on Design and Analysis of Longitudinal Surveys, Ottawa: Statistics Canada, 195-204.

[3] Domingo Ferrer, J. and Mateo Sanz, J.M. (2002) "Practical data oriented microaggregation for statistical disclosure control", IEEE Transactions on Knowledge and Data Engineering, Vol. 14, No. 1, 189–201.

[4] Domingo-Ferrer, J., Martínez-Ballesté, A., Mateo-Sanz, J. M., & Sebé, F. (2006). Efficient multivariate data-oriented microaggregation. The VLDB Journal, 15(4), 355-369.

[5] Domingo-Ferrer, J. and Ursula Gonzalez-Nicolas (2010) "Hybrid microdata using microaggregation", Information Sciences, Vol 180, No. 15, pp. 2834–2844.

[6] Gal, Tamas S., et al. "A data recipient centered de-identification method to retain statistical attributes." Journal of biomedical informatics 50 (2014): 32-45.

[7] Hansen, S.L. and Mukherjee, S. (2003) "A polynomial algorithm for optimal univariate microaggregation", IEEE Transactions on Knowledge and Data Engineering, Vol.15, No. 4, pp. 1043–1044.

[8] Hundepool, A., Van de Wetering, A., Ramaswamy, R., Franconi, L., Capobianchi, A., DeWolf, P., Domingo-Ferrer, J., Torra, V., Brand, R., Giessing, S. (2005): μ-ARGUS version 4.0 Software and User's Manual, Statistics Netherlands, Voorburg NL, http://neon.vb.cbs.nl/casc.

[9] Kabir, Md Enamul, and Hua Wang (2011) "Microdata protection method through microaggregation: A median-based approach." Information Security Journal: A Global Perspective 20.1: 1-8.

[10] L. Sweeney. k-anonymity: A model for protecting privacy. In International Journal on Uncertainty, Fuzziness and Knowledge-based Systems, 2002.

[11] Laszlo, M. and Mukherjee, S.(2005) "Minimum spanning tree partitioning algorithm for microaggregation". IEEE Transactions on Knowledge and Data Engineering,17(7):902–911.

[12] Lin, J. L., Wen, T. H., Hsieh, J. C. and Chang, P. C. (2010) "Density-based microaggregation for

statistical disclosure control", Expert Systems with Applications, Vol. 37, No. 4, pp. 3256–3263.
[13] Lindell, Y., and Pinkas, B. Privacy preserving data mining. In Proc. Int'l Cryptology Conference (CRYPTO), 2000.
[14] Mohammad Naderi Dehkordi, Kambiz Badie, Ahmad Khadem Zadeh. A Novel Method for Privacy Preserving in Association Rule Mining Based on Genetic Algorithms. Journal of software, vol. 4, no. 6, August 2009.
[15] Oganian, A. and Domingo-Ferrer, J. (2001) "On the complexity of optimal microaggregation for statistical disclosure control", Statistical Journal of the United Nations Economic Comission for Europe, Vol. 18, No. 4, 345–354.
[16] Sánchez, David, et al. "Utility-preserving differentially private data releases via individual ranking microaggregation." Information Fusion 30 (2016): 1-14.
[17] Solanas, A. and Martinez-Balleste, A. (2006) "V-MDAV: A multivariate microaggregation with variable group size", in Computational Statistics COMPSTAT 2006, Springer's Physica Verlag, pp. 917–925.
[18] Solanas, A., Martinez-Balleste, A., Mateo-Sanz, J. M., & Domingo-Ferrer, J. (2006, September). Multivariate microaggregation based genetic algorithms. In Intelligent Systems, 2006 3rd International IEEE Conference on (pp. 65-70). IEEE.
[19] Solanas, A., Sebé, F. and Domingo-Ferrer, J. (2008) "Micro-aggregation-based heuristics for p-sensitive k-anonymity: one step beyond". In Proceedings of the 2008 International Workshop on Privacy and Anonymity in Information Society, pp. 61–69, New York, NY, USA, ACM.
[20] Soria-Comas, J., Domingo-Ferrer, J., Sanchez, D., & Martinez, S. (2015). t-closeness through microaggregation: Strict privacy with enhanced utility preservation. IEEE Transactions on Knowledge and Data Engineering, 27(11), 3098-3110.