# A Survey on Methods, Attacks and Metric for Privacy Preserving Data Publishing

Kiran P
Research Scholar
VTU, Belgaum
Karnataka, India

Kavya N P, PhD.
Prof & Head
Dept of MCA, RNSIT
Bangalore,Karnataka,India

## ABSTRACT

Privacy Preserving is a prerequisite for most of the existing systems. Data is usually distributed in the system so the main job of Data Publisher is to retrieve information from different location and to transform it in to some standard format suitable for Data Recipient. This information contains sensitive data which must be preserved by Data Publisher before it is published. So the core of this method is to preserve the sensitivity of data pertaining to individual or company related data. The complexity of its representation and the prerequisite of the current industry have driven lot of research in this direction. In this paper, we provide a review of various methods for anonymization and analyze various disclosures that may happen in each of them. We have also discussed various attacks that may take place during anonymization. A comprehensive study of various metric used for measuring anonymity has also been discussed.

## Keywords

Privacy Preserving Data Mining (PPDM), Privacy Preserving Data Publishing (PPDP), Anonymization, Data Mining, Metric

## 1. INTRODUCTION

Present industry is focused on retrieving, managing and securing huge amount of data. Data may be in the two different forms Structured Representation and Unstructured Representation. Structured Representations have fixed schema like relational tables. Unstructured may be in the form of high dimensional data like transactional data or text files. This huge amount of data is useful for knowledge-based decision making and statistical analysis which is used for executives to make better assessment. Knowledge-based decision also referred as Data Mining has been successfully used in various domains like Weather Forecasting, Market Prediction, Defense and Medical Analysis. This data is retrieved from different locations in different format and converted in to the representation suitable for Data Warehousing. In the above mentioned model the assumption is that Data Warehouse also called as Data Recipient receives data from multiple Data Publishers. Data Publisher collects data from the actual users and usually is an independent organization. For example Medical Decision system which is based on sharing of knowledge between different hospitals for better improvement of clinical data and clinical decisions. In the above environment Medical Decision System is a Data Recipient and hospitals are Data Publishers. Data publishers share data for mutual benefits or due to policy decisions by the government. Data may also be shared for research purpose.

Data collected by Data Publisher from different individuals contain detailed personal information and disease related data, this sensitive information in the raw form published to Data Recipient openly violates individual privacy. AOL an American online web service released its log containing details of the search made by the individuals for research purpose and was intentional, public release of data meant that the entire community could access the data. These logs had personal identification details, which was used for detection of individuals. Newyork Times was able to locate the details of the individual by cross referencing the details of the log with the phonebook listing. Later on AOL acknowledged its mistake and data was removed. Therefore privacy of the individual is of great concern and has become an important chore of research [1]. Privacy also has become more relevant in the current industry because most of the organization store sensitive information about customers or business related information and this data can be mined or linked with external data bases to retrieve the sensitivity of the individual. The current methods and practices concentrate on policies and procedure to restrict the access of sensitive information in published data by either anonymization or swapping or creating synthetic data. Limitation of all these approaches is that they may result in data loss, data may be distorted greatly, privacy of data is low which will impact the efficiency of mining algorithm. There is a trade of between data utility and privacy, if data utility is high then privacy is low and vice versa.

A chore task is to develop methods which publish data in a more unfriendly environment, so that the data remains practically useful without revealing sensitive information. These methods are called as Privacy Preserving Data Mining (PPDM). In past few years research community have proposed different methods and techniques in order to perform PPDM. It has also been discussed in other areas like data base area, statistics and cryptography. Contribution to this research have also come from other areas also like statistics, social science and economics An initial survey on different methods can be found on [2]. There are various PPDM directions adopted to avoid disclosure of sensitive information some of them are Privacy Preserving Data Publishing (PPDP), Query Auditing, cryptographic methods and changing Mining Results. Earlier approaches to privacy were concentrated on anonymization of data based on Data mining algorithm but this proved to be difficult since method applied by the Data Recipient could not be predicted. This paper mainly focuses on PPDP approach to PPDM. In the past research there is no clear distinction between PPDM & PPDP and sometimes referred vice versa.

### 1.1 Architecture of PPDP

PPDP can be represented in the form of layered architecture as shown in figure 1 where the lower layer contains Data Publisher and top layer contains Data Recipient. The overall execution can be divided as data collection phase and data publishing phase. In data collection phase the actual data is collected from record owners by Data Publisher. Data Publisher in turn modifies the data suitable for Data Recipient in a way which ensures privacy this phase is called publishing phase.

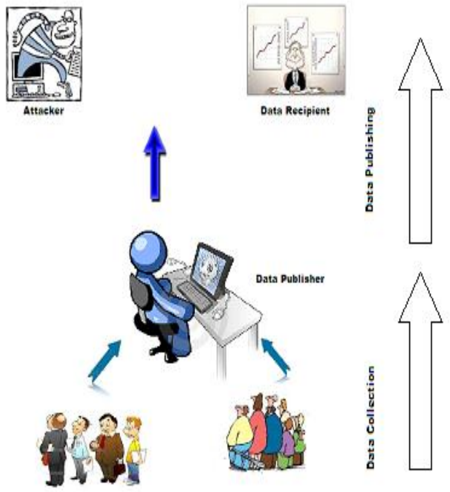

**Fig 1: Architecture of PPDP**

The assumption of this model is that the data publisher is a trusted one and the record owners are ready to share their sensitive information. The Data recipient is untrusted so sensitive data must be protected.

## 1.2 Outline

The focus of survey is to give various Privacy techniques that are available, different algorithms that have been used and the demerits of each of these have been discussed. We have also mentioned different Metrics used for measuring privacy and data utility. Rest of the paper has been organized as follows. In section 2, Classification of Privacy Preservation Techniques has been discussed indicating its representation. Anonymization is the core of this survey, different methods, different types of disclosure and attacks in each of these methods has been discussed in section 3. In section 4, we have indicated important anonymization algorithms that are required to implement anonymization methods. Research is oriented towards metric based analysis, different metric is required for measuring privacy failure and data utility this has been discussed in section 5. We state our conclusion and future directions in section 6.

# 2. CLASSIFICATION OF PRIVACY PRESERVING TECHNIQUES

The main idea of PPDP is to develop methods such that the sensitivity of data is not released. A number of effective techniques have been formalized in literature most of them tend to modify the original data in order to retain the sensitivity of data. PPDP has been classified into following categories

## 2.1 Randomization method

It is the process of adding noise to the original data in order to mask attributes from disclosure [3,4]. There are different ways of Randomization the simplest is additive randomization and can be described as follows. Let X={x1,x2, . . .,xn} be a set of data records. For each xi an element of X, noise is added which is taken from probability distribution f(y) and are denoted by y1,y2,… , yn. the resultant distribution can be represented as x1+y1,x2+y2,…,xn+yn. various techniques of randomization were proposed in literature[5,6]. The accuracy of privacy preservation depends on how large the distribution y would be and the right amount of randomization. Randomization can be done either by adding or multiplying noise [7]. One of the disadvantages is that results are approximate and has huge information loss.

## 2.2 Data Swapping

It is a method in which values of records are swapped which maintains the statistical inference of the relation in order to preserve privacy [8].

## 2.3 Cryptographic approach

Revolution of communication via internet has forced several areas one such is Distributed Data Mining. In this environment Data is distributed in multiple sites and in order to mine the data must be securely retrieved [9,10,11] . This approach is advantageous for two different reasons first there are lots of algorithm to implement cryptographic methods and it is a well defined model for privacy.

## 2.4 Anonymization Approach

The most common approach to preserve sensitivity is to modify the contents of the record owners before publishing the data this approach itself is called Anonymization. Basic form of relational schema which is used by Data Publisher can be represented as

**R(U_ID,Q_ID_1,Q_ID_2,…,NQ_ID_1,NQ_ID_2,…,SV_1, SV_2,..)**

Where U_ID indicates user identification and are explicit values that can be directly used for inferring the identity of the individual. For example Social Security Number can be used to retrieve information regarding a person in USA. Q_ID are quasi identifiers which can used by attackers to link this value to an external data base to retrieve the identity of the individual. For example gender, ZIP code and DOB can be used with external data base like voters list to identify the person. Pseudo SQL query may be as follows

```
SELECT *
FROM VOTERS_TABLE AS V
WHERE V.ZIP=’&ZIP’ AND
V.GENDER=’&GENDER’ AND V.DOB=’&DOB’;
```

SV contains sensitive value which is used for mining and statistical analysis. In medical records sensitive value would be the decease of the patient. NQ_ID are non quasi identifier which does not belong to any of the categories mentioned above. They are published if they are relevant for data mining. To prevent the disclosure of information attacker will modify the relation R to Rl

**Rl(Q_ID_1,Q_ID_2,…,NQ_ID_1,NQ_ID_2,…,SV_1, SV_2,..)**

In Rl U_ID is removed and Q_ID are anonymized such that it satisfies the privacy model representation and ensures confidentiality. In this paper we have concentrated more on anonymization approaches to privacy preservation.

# 3. ANONYMIZATION METHODS AND PRIVACY ATTACK

In this section, we look at different representation of anonymization and also understand how privacy attack may take place in each of these algorithms. Privacy attacks can be broadly classified in to two categories, the first category occurs when an attacker can link the published record to an external data base and can infer sensitive information. In second category the attacker must have extra knowledge to infer the sensitive information. First category is usually referred as record linkage, attribute linkage and table linkage and the assumption is that the attacker knows details of the person sensitive value is present in the released table. The second category comes under probabilistic linkage.

## 3.1 k-Anonymity

is a property that avoids possible re-identification of the respondents from published data[12,13]. For example let us consider a published table where U_ID are removed and is of the form

**T(QID_AGE,QID_SEX, QID_ZIP,S_ DISEASE )**

In this values of the released attributes like ZIP code & Age can also be present in external public data bases like voters list which can be used to re-identify individuals there by sensitivity of data can be obtained. To understand this problem let us consider table 1 containing information of released voters data.

**Table 1. Released voters data**

| Name | Age | Sex | Zipcode |
|---|---|---|---|
| Arun | 25 | Male | 53711 |
| Sita | 28 | Female | 55410 |
| Sarasa | 31 | Female | 90210 |
| Chetan | 26 | Male | 53711 |
| Zita | 27 | Female | 53712 |

**Table 2. Released Medical Data**

| Name | Age | Sex | Zipcode |
|---|---|---|---|
| Arun | 25 | Male | 53711 |
| Sita | 28 | Female | 55410 |
| Sarasa | 31 | Female | 90210 |
| Chetan | 26 | Male | 53711 |
| Zita | 27 | Female | 53712 |

Table 2 contains released medical data. In this he record <25,Male,53711,Heart> of the released data can be linked to unique record <Arun, 25,Male,53711> of voters list there by revealing Arun disease to be Heart related this type of attack is called Linking attack and the disclosure is called Record disclosure.. To overcome linking attacks Samarati and Sweeney proposed a definition of privacy called k-anonymity[12,14] and the definition

Definition 1 (k-anonymity) A table satisfies k-anonymity if every record in the table is indistinguishable from at least k − 1 other records with respect to every set of quasi-identifier attributes; such a table is called a k-anonymous table.

In other words each group of quasi identifier values must have at least k-1 records and can be cheeked by linking a record in released data to multiple records publicly available data base. Two main methods that are available for enforcing k-anonymity on published data are generalization and suppression. Generalization or Suppression is a technique which either modifies or hides the contents of quasi identifiers. For a labeled attribute, a specific value can be modified to a general value according to a predefined hierarchy. This hierarchy in generalization corresponds to a domain generalization hierarchy and a corresponding value generalization hierarchy on the values in the domains. For example zip code can have a hierarchy as mentioned in figure 2c. 53711 can be generalized to 5371* similarly 5371* can be generalized to 537** and so on with respect to the hierarchy. Similar to labeled attributes for a number, exact values can be replaced with an interval. For example all values in the range 20 to 50 can be generalized to <50 or [20-50] label. There are various ways of defining hierarchy all of which concentrates on privacy and data usefulness and exploring search space. The turnaround approach is called specialization. Some of the important generalization methods that are used in literature are Full Domain generalization [15], sub-tree generalization [16,17,18], cell generalization [19] and Multi dimensional generalization scheme[20].

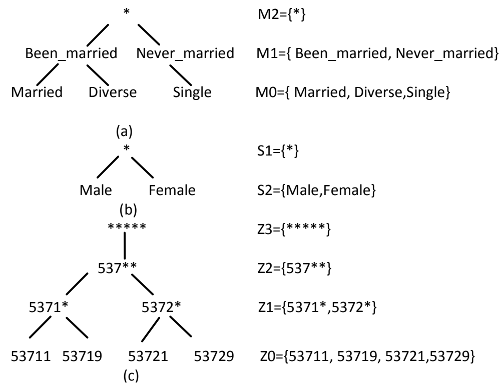

**Fig 2: An example of value and domain generalization hierarchy**

In full domain generalization all values are generalized to the same level in the hierarchy. For example, if 53711 is generalized to 5371* then 53729 is generalized to 5372* and so on. In sub-tree generalization some of sub branches are generalized and some will be as it is. Above mentioned methods are called global recoding. In cell level based on the requirements some records which have 53711 will be generalized to 5371* and some will remain as it is. It is based on individual data values so it is called local recoding. All the above mentioned methods are also called single-dimensional generalization. Multi dimensional generalization is a mehod in which entire record is replaced with another record. Let Dom(i) be the domain of an attribute Attr(i). single-dimensional can be defined as f(Dom(i))→ Dom_h , for each attribute Attri(i) in quasi identifiers. Multidimensional changes every record f(Dom_Attr(1), Dom_Attr(2),…, Dom_Attr(n))→ Dom_Attr_h(1), Dom_Attr_h(2),…, Dom_Attr_h(n). In summary Full Domain generalization has the smaller search space as compared with other methods. Since each value must be generalized to the same level in the hierarchy data distortion is larger. Cell based method has a lesser data distortion but data utility is less. There are two widely used suppression methods Record suppression [18] and Value suppression. In record suppression the record is removed from the anonymized table if the no of records is less that a predefined threshold. Value suppression refers to suppression of only few values in the quasi attributes and is done by using a predefined threshold. For example the result of 2-anonymus table is shown in table 3.

**Table 3. 2-anonymus Medical Data**

| Age | Sex | Zipcode | Disease |
|---|---|---|---|
| [20-30] | Male | 5371* | Heart |
| [20-30] | Male | 5371* | Heart |
| [30-40] | Male | 5371* | Broken Arm |
| [30-40] | Male | 5371* | Hang Nail |
| [20-30] | Female | 5371* | Hepatitis |
| [20-30] | Female | 5371* | AIDS |

Variations of k-anonymity exists in literature (X,Y)-anonymity[21] and MultiR k-anonymity[22]. (X,Y)-anonymity specifies that each value on X is linked to at least k distinct values on Y and this concept was motivated by sequential releases. Data base is made up of multiple tables author [22] proposed anonymization methods for a collection of tables rather than a single table.

## 3.2 Attacks on K-anonymity

K-anonymity overcomes record linkage but does not overcome attribute disclosure. For example let us consider table 3 which contains 2-anonymus Medical Data. By observing first group of records it is difficult to find whether <[20-30], Male, 5371*, Heart> is linked to <Arun, 25, Male,53711> because it can also be linked to < Chetan, 26, Male,53711> so it overcomes linking attack There are two important attacks that take place in k-anonymity. i) Homogeneity Attack ii) Back Ground Knowledge Attack

Homogeneity Attack: occurs when the entire QID group has the same sensitive values. Suppose the attacker knows arun age to be 25 living in zip 53711 and table 3 is available, attacker can confidently conclude that arun has heart disease this is called Homogeneity Attack. Since the quasi group of 5371* has only one sensitive value which is heart disease.

Background Knowledge Attack: occurs when the attacker has a back ground knowledge of the sensitive attribute. For Back ground knowledge attack, suppose the attacker knows zita age is 27 and zip is 57312, attacker can conclude zita corresponds to the last group in table 3. Further, suppose the attacker knows that 57312 is a region where sex workers are more. This back ground knowledge enables the attacker to conclude zita most likely has AIDS.

## 3.3 l-Diversity

To overcome the limitations of k-anonymity Machanavajjhala [23] introduced l-diversity. The definition of l-diversity as specified by [23] and its variant

Definition 2 ( l-diversity Principle) An equivalence class is said to have l-diversity if there are at least l "well-represented" values for the sensitive attribute. A table is said to have l-diversity if every equivalence class of the table has l-diversity.

Author [23] gave a number of interpretations of the term "well-represented" by indicating various properties for formation of the quasi group. Dimensions of l-diversity are i) Distinct l-diversity ii) Entropy l-diversity and iii) Recursive (c, l)-diversity

**Table 4. 2-Diversity Medical Data**

| Age | Sex | Zipcode | Disease |
|---|---|---|---|
| [20-40] | Male | 5371* | Heart |
| [20-40] | Male | 5371* | Heart |
| [20-40] | Male | 5371* | Broken Arm |
| [20-40] | Male | 5371* | Hang Nail |
| [20-40] | Female | 5371* | Hepatitis |
| [20-40] | Female | 5371* | AIDS |

## 3.4 Attacks on l-Diversity

l-Diversity also overcomes record level disclosure but does not completely overcome attribute level disclosure. There are two important attacks that happen in l-Diversity i) Skewness attack and ii) Similarity Attack.

Skewness Attack happens when the attacker can derive the sensitive value based on the frequency distribution of it. For example suppose that a quasi group 2 in table 4 has an equal number of positive & negative records, which satisfies 2-Diversity. However, this indicates a probability of 50% of having the sensitive attribute as compared to the probability of 1% in the original population.

Similarity Attack happens when all the sensitive attribute in a quasi group are distinct but semantically similar. For example ulser, gastric & dyspepsia are the sensitive attribute of a quasi group which satisfies 3-Diversity. However, all the above mentioned sensitive attribute are semantically related to stomach disease which can be derived by the attacker.

## 3.5 t-Closeness

To overcome the disadvantages of l-Diversity Ninghui Li [23] introduced t-Closeness. In this method privacy is measured by attackers information gain about the sensitive attribute. Attacker, before observing the anonymized data has some prior belief based on the distribution of sensitive attribute in publicly available data, after observing the anonymized data his belief changes to posterior belief. Information gain is measured as the difference between posterior belief and prior belief. In this method, it limits and restricts the extends to which the attacker can infer additional information about individuals. This is achieved by making the distribution of sensitive values in the publicly available data base equal to the distribution of sensitive value in each quasi group. As specified by [23] the definition is

Definition 3 (The t-closeness) An equivalence class is said to have t-closeness if the distance between the distribution of a sensitive attribute in this class and the distribution of the attribute in the whole table is no more than a threshold t. A table is said to have t-closeness if all equivalence classes have t-closeness. t-closeness uses the Earth Mover Distance (EMD) function to measure the sensitive attribute frequency distribution of publicly available data with the distribution of qasi group and requires the closeness to be within t.

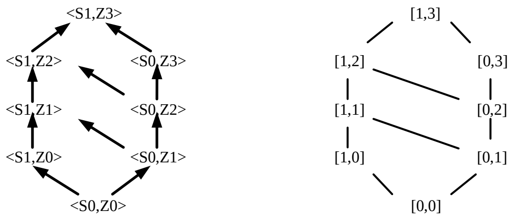

**Fig 3: Hierarchy DGH<S0Z0> and corresponding Hierarchy of distance vectors**

## 3.6 Limitations of t-Closeness

There is no standard procedure to enforce t-closeness. Each attributes are generalized independently. Different protection levels cannot be specified for sensitive attributes. Attribute linkage cannot be prevented on numerical sensitive attributes [26]. Most important disadvantage is that it greatly degrades the data utility because it requires the distribution of sensitive values to be the same in all qid groups.

# 4. ALGORITHMS FOR ANONYMIZATION

There are different algorithms proposed in literature [12,15,25]. These algorithms were initially designed for checking the property of k-anonymity. Same algorithms can also be used for checking l-divercity, t-closeness and its variants. The initial algorithm for implementing k-anonymity was discussed in [12] which uses binary search method for finding optimal anonymization.

## 4.1 Samarati's Algorithm

It is based on Attribute Generalization and Tuple suppression. It requires value and domain generalization hierarchy to be created. For example let us assume the value and domain hierarchy of gender and zipcode as indicated in figure 2(b) & 2(c) and its distance vector can be shown in figure 3. Distance vector indicates how attribute generalization can be made. Height of VHD in figure 3 is 4.first index indicates gender and second index indicates zipcode. For example the record input is < Male, 53721 > and if the index is [1,2] the resultant generalization would be < *,537** >. Index of [0,0] is the actual records which are not generalized and [1,3] indicates highest level of generalization. The algorithm as stated by [12] is indicated in figure 4.

```
Algorithm 1(Samarati's)
INPUT: Table T containing data with qasi identifiers  Q to be generalized, value k which indicates number of minimum values in quasi group , suppression threshold S, Value domain hierarchy VDH of the distance vectors corresponding to the domain generalization hierarchy DGH, where t is the tuples of the domains of the quasi-identifier attributes.
OUTPUT: The distance vector of a generalized table vec and its generalization $T^*$
METHOD: Executes a binary search based on VDH height .
1.low =0; high=height(⊤, VDH ); sol= ⊤;
2.while low< high
3.index = (low+high)/2;
4.Create_Vectors= {vec | height(index,VDH ) = try}
5.Reach_k= false
6.while Create_Vectors != ∅ ∧ reach_k == false
6.1 Select and remove a vector vec from        Create_Vectors
6.2 if satisfies_k_anonymity(vec,k,$T_i$,S)
    then sol= vec; reach k= true;T*=$T_i$;
    6.3if reach k = true  then high= index;
        else low= index + 1;
7.Return sol,$T^*$
```

**Fig 4: Samaratis algorithm**

Function height(index,VDH) retrieves a vector of index values. For example height(2,VDH) returns {[1,1], [0,2] } and so on. Satisfies_k_anonymity checks whether each quasi group satisfies k-anonymity property.

## 4.2 k-Optimize

Bayardo and Agrawal [25] proposed an Attribute Generalization and tuple Suppression algorithm called k-Optimize. This method uses set-enumeration and tree based search strategy for finding optimal generalization. It assumes that there is a predefined ordering of values in Q attributes and associates an index value. This index is used for anonymization of tuples in the public data base. For example consider the patient data in Table 2 with Q={Age,Sex,Zipcode} and suppose Age precedes sex that, in turn precedes Zipcode and the order among values inside each attribute domain is [20-30],[30-40], [40-50] for age, [M],[F] for sex and [53711],[53719],[53721], [53729] for zipcode. Figure 5 represents the index assignment obtained when no generalization is applied.

| | | | | |
|---|---|---|---|---|
| [20-30] | [30-40] | [40-50] | [M] | [F] |
| 1* | 2 | 3 | 4* | 5 |
| [53711] | [53719] | [53721] | [53729] | |
| 6* | 7 | 8 | 9 | |

**Fig 5: Index assignment to attributes Age, Sex and Zipcode**

Generalization of a set I containing index values is equal to the union of individual index values. For example union of index 1 & 2 means that age values [20-30] and [30-40] will be generalized to [20-40]. This approach uses the union of index values taking in to account the least index in each of the attributes. For example the least index value in age 1, sex is 4 and zipcode is 6 which have been marked with * in figure 5.

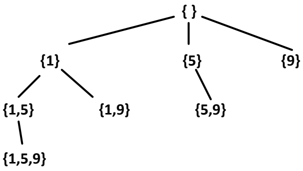

**Fig 6: An example of set enumeration tree over a set I={1,5,9}**

This indicates that given a set I of index value, if a particular index does not appear in I, then it has been generalized to the nearest index. For example if the set I= {1, 5, 9} indicates that index 2 & 3 has been generalized to the same value as 1, 7 & 8 to 6. Since the least value in any attribute will appear in generalization, it can be removed form I. the resultant set can be represented as I={5,9}. K-optimize algorithm builds' a set enumeration tree over the set I , contains all the possible subset I. figure 6 represents a set enumeration of I={1,5,9}. Each node in the tree indicates how T is generalized. The empty set {} indicates the highest generalization and I={1,5,9} indicates the most specific generalization. Algorithm visits each node in set enumeration tree , checks k-anonymity and also computes the cost metric for that node. This cost metric is compared with the earlier best cost value that is computed, if the value is lesser than the previously computed value then this node becomes I* and this cost metric will become the best cost. Visiting the complete tree is not optimal so k-optimize proposes a heuristic method for removing a node and its subsequent branches. A node and its descendants can be removed if that node cost is not optimal. The algorithm as specified by [25] is shown in figure 7. k-Optimize computes the best solution among the entire enumeration tree.

```
Algorithm 2 (k-Optimize)
INPUT: I set of index values corresponding to the original domains of T [Q] and k for anonymity
OUTPUT: $I^*$ set of index values representing the k-anonymous generalized table
METHOD: Execute a breadth search on the enumeration tree starting from tree
1. root = { }
2. best_cost = infinity
3. best_sol = null
4. Optimize(root,best)
OPTIMIZE(node,best)
4.1.if Satisfy(node) then current_cost= Cost(node);
4.2. if current_cost ≤ best_cost then
/*initialization*/
best_cost = current_cost
best_sol = node
4.3.for each i∈ {idx|idx ∈ I ∧
    (∀j ∈ node, idx ≥ j ∨ node = ∅)}
    4.3.1. child= node ∪ {i}
    4.3.2. lb= LowerBound(child)
    4.3.3. if lb≤best_cost then
      best= Optimize(child,best)
      else Prune(node) /* prune nodes having node as a subset */
 4.4.return(best)
```

**Fig 7: k-Optimize algorithm**

## 4.3 Incognito

LeFevre, DeWitt and Ramakrishnan [15] proposed an efficient algorithm for computing k-minimal generalization, called Incognito, which takes advantage of a bottom-up aggregation along dimensional hierarchies and a priori aggregate computation. The core idea behind this method is that if a group contains Q quasi identifiers is k-anonymous then Q* is also k-anonymous if $Q^* \subset Q$. the domain generalization hierarchy is traversed by using breadth search and checked for k-anonymity property. In the first iteration k-anonymity is checked for single attribute of Q, removing those nodes which does not satisfy k-anonymity. In the next iteration, it combines the remaining generalization in pairs and checks the same for k-anonymity, then for triple values and so on. Until the entire set of Q is considered. In the given Domain Generalization Hierarchy, if one of the node satisfies k-anonymity then all its directed generalization will also satisfy k-anonymity and therefore they will not be taken for consideration. For example in Table 1 let us consider quasi attribute is Q={sex, zipcode} and k=2. In the first iteration incognito checks 2-anonymity on single attribute S0 & Z0. S0 satisfies 2-anonymity but Z0 doesn't , so the next generalization of Z0 that is Z1 is cheked for anonymity. Z1 satisfies 2-anonymity. In the next iteration it checks for the pair <S0,Z1> satisfies 2-anonymity so its directed generalization also satisfy 2-anonymity.the resultant hierarchy computed may be as shown in figure 8.As specified by [15] the algorithm is indicated in figure 9.

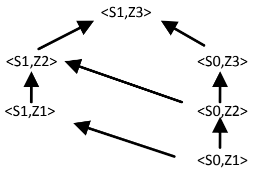

**Fig 8: Resultant Incognito Generalization for gender and zipcode**

```
Algorithm 3 (Incognito)
Input: A table T to be k-anonymized, a set Q of n quasi-identifier, and a set of dimension tables (one for eachquasi-identifier in Q)
Output: The set of k-anonymous full-domain generalizations $T^*$
$Currenr\_node_1$ = First Node in the domain generalization hierarchies of attributes in Q
$Edge_1$= Edges for $Currenr\_node_1$in the domain generalization hierarchies of attributes in Q
1. q = an empty queue
2. for i = 1 to n do
/* Currenr_node i and Edge i define a graph of generalizations*/
3.Si = copy of Currenr_node i
4.{roots} = all nodes to Currenr_node i with no edge to $Edge_i$ directed to them
5.Insert {roots} into queue, keeping queue sorted by height
6.while q is not empty do
6.1. node = Remove first item from q
6.2. if node is not marked then
6.2.1.if node is a root then freq_Set = Compute frequency set of T with respect to attributes of node using T.
    else  freq_Set = Compute frequency set of T with respect to attributes of node using parent's frequency set.
6.3. Use freq_Set to check k-anonymity with respect to attributes of node
6.4. if T is k-anonymous with respect to    attributes of node then Mark all direct generalizations of node
else Delete node from $S_i$ Insert direct generalizations of node into q, keeping q ordered by height
7. $Currenr\_node_i$ =$Currenr\_node_i$+1;
8. $Edge_i$=$Edge_i$+1
9. GraphGeneration($S_i$, $Edge_i$)
10.end for
11. return Projection of attributes of Si onto T and dimension tables
```

**Fig 9: Incognito algorithm.**

## 4.4 Mondrian Multidimensional Partitioning Anonymization

Proposed by LeFevre, DeWitt and Ramakrishna[27], this method uses multidimensional global recoding technique. Let us assume that T contains data to be anonymized having Q quasi attributes. the value of Q can be represented as a set of points in a multidimensional space, where each attribute represents one dimension. k-anonymity is achieved by partitioning the given space in to regions which contain at least k points. Each of Q values in every partition is generalized to the same value. The advantage of this method is that each partition can be generalized independently. This relaxed representation has better quality than single level generalization. The author has proposed a greedy algorithm which works as follows. Let us assume that the region r is generated based on the current quasi attribute value. In every iteration the algorithm chose's a dimension d and divides the region at the median value x considering d such that d>x will belong to one of the region and the remaining to the other region. This division is made if the region contains greater than k points. All the tuples in each region is generalized based on predefined hierarchy. Algorithm as stated by [27] is shown in figure 10.

```
Algorithm4(Mondrian: Multidimensional Partitioning Anonymization)
Input: partition P
Output: a set of valid partitions of P
1.If no allowable multidimensional cut for P then
1.1 return P
else  1.2. dim = choose_dimension(P)
1.3. fs = frequency_set(P; dim)
1..4. splitVal = find_median(fs)
1.5. lhs = {t ∈ partition: t.dim ≤ splitVal}
1.6. rhs ={ t ∈ partition : t.dim  > splitVal}
1.7.return partition_anonymize(rhs)U partition anonymized(lhs)
```

**Fig 10: Mondrian Multidimensional Partitioning Algorithm**

For example consider Q={age, zipcode} and k=2 figure 11(a) represents the two dimensional representation of the table 2 for age and zipcode, where occurrence of a value is represented as a point. Suppose the division of region is initially made on age the resultant is shown in figure 11(b). In the next iteration division can be made on age or zipcode, in our example we have taken zipcode. The resultant partition is shown in figure 11(c).

## 5. METRICS FOR MEASURING HIDING FAILURE AND DATA QUALITY

The core of most PPDP algorithms is that they either modify or block the data values in order to hide sensitive information. Such methods can be measured with two important parameters, first is by its hiding sensitive data and second is by measuring data quality of the transformed data. More the amount of transformation More the sensitivity but the quality of data is less. Therefore both hiding failure and Data quality metric are very much important in evaluating PPDP techniques.

## 5.1 Hiding Failure(HF)

Hiding failure parameter is used to measure the amount of information that can be derived after the data has been modified. The main goal of privacy preserving algorithm is to have zero hiding failure , this leads to more amount of information loss. Thus , some PPDP algorithms have also been designed which allow the data publisher to choose the amount of sensitive data to be hidden. In [28] author has defined Hiding Failure(HF) as the percentage of restrictive patterns that are discovered from the sanitized database and is measured as

$$\mathbf{HF}=\frac{\boldsymbol{Number\ Restrictive\ Patterns(T*)}}{\boldsymbol{Number\ Restrictive\ Patterns(T)}}$$

Where T & T* represents original data set and generalized data set

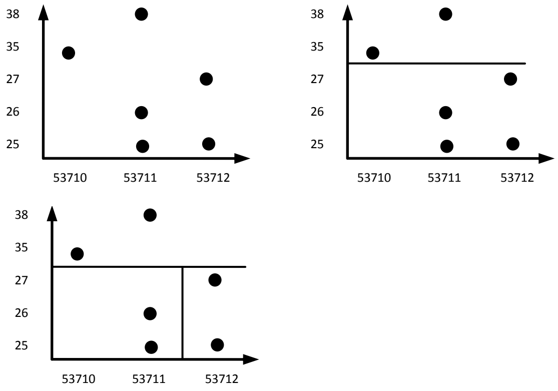

**Fig 11: Region Representation (a) and possible partition (b) & (c)**

## 5.2 Data quality

There are various data quality metrics that have been proposed in literature that are either generic or data specific. Currently there are no standard metric that has been widely accepted by the research community. Data quality can be either measured after PPDP or after Data Mining. The accuracy can be measured by information loss which is a resultant of hiding sensitive data[29]. Less the information loss better is the data quality. Given a original database T with N attributes and r records, if we identify as generalization scheme a domain generalization hierarchy GHT with a depth h, it is possible to measure the information loss (IL) of modified database T as

$\boldsymbol{Iloss(r)} = \sum_{vg\in r}(\boldsymbol{wi} * \boldsymbol{Iloss(vg)})$ Where wi –penalty weight of attribute Ai of vg $\boldsymbol{Iloss(vg)} = \frac{\boldsymbol{ABS(vg)-1}}{\boldsymbol{ABS(DA)}}$

Where vg is number of descendents in Value Generalization Hierarchy and DA is the number of domain values in the attribute. Iloss for the entire T* is given by

$$\boldsymbol{Iloss(T^*)} = \sum_{r\in T} \boldsymbol{Iloss(r)}$$

Classification Metric (CM), is introduced by Iyengar [18] to optimize a k-anonymous dataset. It is defined as the sum of the individual penalties for each row in the table normalized by the total number of rows r.

$$\boldsymbol{CM(T^*)} = \frac{\sum_{all\ rows} \boldsymbol{penalty(tuple\ t)}}{\boldsymbol{r}}$$

Penalty value is calculated based on wither tuple t is suppressed or generalized. If the tuple is not changed then its penalty is zero. Penalty value of 1 is taken if the tuple is suppressed or generalized. It can be used basically for classifying over generalized data.

Another interesting metric is the Discernibility Metric(DM) proposed by Bayado and Agrawal [25]. This discernibility metric calculates the cost by charging a penalty to each tuple for being indistinguishable from other tuples. Let t be a tuple from the original table T, and let GT*(t) be the set of tuples in an anonymized table T* indistinguishable from t or the set of tuples in T* equivalent to the anonymized value of t. Then DM is defined as follows

$$\boldsymbol{DM(T^*)} = \sum_{t\in T} |\boldsymbol{G_{T*}(t)}|$$

In many situations, suppressions are considered to be most expensive in the sense of information loss. Thus, to maximize data utility, tuple suppression should be avoided whenever possible.

Minimal distortion(MD) proposed by Samarati [12] is based on charging penalty for each value which is generalized or suppressed. Each hierarchy is assigned a penalty when it is generalized to the next level with in the domain generalization hierarchy. For example a penalty of 10 units is taken when generalizing 53711 to 5371* another 10 units for generalizing 53711 to 537** and so on.

In certain requirements there is a necessity to measure the data mining results after anonymization, this kind of metric emphasis on how data is used. This is also dependent on the knowledge that can be derived from the original data set. Let us assume that the data is used for clustering then the information loss can be measured as the percentage of points that have changed against the original classification. As in [30], a misclassification error(ME) is defined as

$$\boldsymbol{ME} = \frac{\mathbf{1}}{\boldsymbol{NP}} \sum_{i=1}^{k} (\,|\mathbf{cluster_i(T)}| - |\mathbf{cluster_i(T^*)}|)$$

where NP represents the number of points in the original dataset, k is the number of clusters under analysis, and |Clusteri(T)| and |Clusteri(T*)| represent the number of data points for the ith cluster in the original dataset T and the generalized dataset T* respectively. Anonymization technique tries to either suppress or modify the existing values there by affecting the clustering that could have been formed. This loss must be minimal as it affects the mining resultant.

Data usage can also be measured by distinguishing between Lost information and Artifactual information. Lost information represents the percentage of patterns that are suppressed and he Artifactual information represents the percentage of patterns created by the modification due to anonymization technique. Oliveira and Zaiane [28] defined two metrics misses cost and artifactual pattern which is related to lost information and artifactual information. Misses cost measures the number of patterns missed. This happens when the paterrn of a particular cluster loses its support due to the modification. The misses cost (MC) is computed as follows

$$\boldsymbol{MC} = \frac{\boldsymbol{No\ Of\ Patterns\ (T) - No\ Of\ Patterns(T^*)}}{\boldsymbol{No\ Of\ Patterns\ (T)}}$$

## 6. CONCLUSION AND FUTURE DIRECTION

Knowledge based retrieval has given rise to association of data from different sources which are distributed across different locations in different format. Since the data is retrieved from different Data recipient the sensitivity disclosure problem about individual or company exists. This has given rise to a new research direction called Privacy Preserving Data Publishing. In this survey, we have presented an overview of different methods used in protecting sensitive data and analyses of the existing algorithms available and indicate the various disclosures & drawbacks in each of them. In particular, we have focused more on Anonymization Techniques used for Privacy Preserving Data Publishing and we have also mentioned Metric used for measuring hiding failure & data quality.

There are several future research directions along the way of analyzing different PPDP algorithm and its application. The main challenges in PPDM are how to have minimum generalization of data such that there is maximum utility. There is also a requirement for a common framework which overcomes different disclosure and attacks. During anonymization most data are not sensitive so generalization of the entire data is meaningless; research in this direction is to be considered. There is also a requirement to develop a comprehensive architecture which combines data publisher and data recipient. In distributed environment efficiency will pay an important role, so an efficient algorithm which tries to balance between sensitive disclosure, data utility and communication cost is required.

## 7. REFERENCES

[1] Han Jiawei, M Kamber. Data Mining: Concepts and Techniques, Beijing: China Machine Press, 2006, pp.1-40.

[2] Verykios V S, Bertino E, Fovino I N, Provenza L P, Saygin Y, Theodoridis Y. State-of-the-art in privacy preserving data mining, ACM SIGMOD Record, 2004.

[3] Agrawal D. On the Design and Quantification of Privacy- Preserving Data Mining Algorithms, ACM PODS Conference, 2002.

[4] Warner S L. Randomized response: A survey technique for eliminating evasive answer bias. Journal of the American Statistical Association,1965.

[5] Zhang P, Tong Y, Tang S, Yang D. Privacy-Preserving Naive Bayes Classifier, Lecture Notes in Computer Science, 2005, Vol 3584.

[6] Zhu Y, Liu L. Optimal Randomization for Privacy-Preserving DataMining, ACM KDD Conference, 2004.

[7] Agrawal R, & Srikant R. Privacy preserving data mining, Proceedings of ACM SIGMOD Conference on Management of Data (SIGMOD'00), Dallas, 2000.

[8] Fienberg S,McIntyre J. Data Swapping: Variations on a Theme by Dalenius and Reiss, Technical Report, National Institute of Statistical Sciences, 2003.

[9] Pinkas B. CryptographicTechniques for Privacy-PreservingDataMining, ACM SIGKDD Explorations, 2002.

[10] Laur, H Lipmaa, and T Mieliainen. Cryptographically private support vector machines, In Twelfth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2006, pp. 618-624.

[11] Ke Wang, Benjamin C M, Fung and Philip S Yu. Template based privacy preservation in classification problems, In ICDM, 2005, pp. 466-473.

[12] Pierangela Samarati. Protecting respondents' identities in micro-data release, IEEE Transactions on Knowledge and Data Engineering, November 2001.

[13] Pierangela Samarati and Latanya Sweeney. Generalizing data to provide anonymity when disclosing information. In Proc. Of the 17th ACM-SIGMOD-SIGACT-SIGART Symposium on the Principles of Database Systems, Seattle, WA, 1998, pp. 188.

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

[15] Lefevre K, Dewitt D J and Ramakrishnan R. Incognito: Efficient full-domain k-anonymity, In Proceedings of ACM SIGMOD. ACM, New York, 2005.

[16] Fung B C M, Wang K and Yu P S. Anonymizing classification data for privacy preservation, In EEE Trans. Knowl. Data Engin, 2007, v. 19, pp 711–725.

[17] Fung B C M, Wang K and Yu P S. Top-down specialization for information and privacy preservation, In Proceedings of the 21st IEEE International Conference on Data Engineering (ICDE), 2005, pp 205–216.

[18] Iyengar V S. Transforming data to satisfy privacy constraints, In Proceedings of the 8th ACMSIGKDD. ACM, New York, 2002, pp. 279–288.

[19] Xu J,Wang W, Pei J,Wang X, Shi B and Fu A. W. C. Utility-based anonymization using local recoding, In Proceedings of the 12th ACM SIGKDD Conference. ACM, New York, 2006.

[20] Lefevre K,Dewitt D J and Ramakrishnan R, Workload-aware anonymization, In Proceedings of the 12th ACM SIGKDD. ACM, New York, 2006

[21] Wang K and Fung B C M, Anonymizing sequential releases, In Proceedings of the 12th ACM SIGKDD Conference. ACM, New York, 2006.

[22] Nergiz M E, Clifton C and Nergiz A E, Multirelational k-anonymity, In Proceedings of the 23rd International Conference on Data Engineering (ICDE), 2007, pp. 1417–1421.

[23] Machanavajjhala A, Gehrke J, Kifer D and Venkitasubramaniam M. l-diversity: Privacy beyond k-anonymity, In Proceedings of the 22nd IEEE International Conference on Data Engineering(ICDE), 2006.

[24] Ninghui Li , Tiancheng Li , Suresh Venkatasubramanian. t-Closeness: Privacy Beyond k-Anonymity and l–Diversity ICDE Conference, 2007.

[25] R J Bayardo, R Agarwal. Data privacy through optimal k-anonymization, In Proc. of the 21st International Conference on Data Engineering(ICDE'05), Tokyo, Japan, 2005, pp 217-228.

[26] Li J, Tao Y and Xiao X. Preservation of proximity privacy in publishing numerical sensitive data, In

Proceedings of the ACM Conference on Management of Data (SIGMOD), 2008, pp. 437–486.

[27] Kristen LeFevre, David J DeWitt and Raghu Ramakrishnan. Mondrian multidimensional k-anonymity, In Proc. of the International Conference on Data Engineering (ICDE'06), Atlanta, Georgia, April 2006.

[28] Oliveira S R M , Zaiane O R. Privacy preserving frequent itemset mining, In: IEEE icdm Workshop on Privacy, Security and Data Mining, 2002, vol. 14, pp. 43–54.

[29] Xiao X and Tao Y. Personalized privacy preservation, In Proceedings of the ACM SIGMOD Conference.ACM, New York, 2006.

[30] Oliveira S R M, Zaiane O R. Privacy preserving clustering by data transformation, In 18th Brazilian Symposium on Databases, 2003, pp. 304–318.