ISSN: 2350-0328

International Journal of AdvancedResearch in Science, Engineering and Technology

Vol. 5, Issue 9 , September 2018

A Survey on Privacy Preserving Data Mining PPDM concepts and methods

Dr. AliaaKarim Abdul Hassan , Hanan Qassim Jaleel

University of technology, Department of Computer Science
Baghdad College of Medical Sciences

ABSTRACT: In recent times data mining has gained immense importance as it paves way for the management to obtain hidden information and use them in decision-making. While dealing with sensitive information it becomes very important to protect data against unauthorized access. Large companies and Organizations face great risks while sharing their data. Most of this sharing takes place with little secrecy. It also increases the legal responsibility of the parties involved in the process. So, it is crucial to reliably protect their data due to legal and customer concerns. Protection of privacy has become an important issue in data mining research for protecting sensitive data or knowledge. A number of privacy-preserving data mining methods have recently been proposed, which take either a cryptographic or a statistical approach.

PPDM can be executed at different stages of the information processing pipeline, such as data collection, data publication, output publication, or distributed data sharing , This paper focuses on the concepts and methods of for privacy preserving data mining which includes PPDM during data collection , PP data publishing , PP data distributing and PP during output results

I. INTRODUCTION

Privacy preserving data mining (PPDM) deals with protecting the privacy of individual data or sensitive knowledge without sacrificing the utility of the data. People have become well aware of the privacy intrusions on their personal data and are very unwilling to share their sensitive information. In current years, the area of privacy has realized fast advances because of the increases in the ability to store data. In particular, recent advances in the data mining field have lead about privacy [2].The aim of privacy preserving data mining(PPDM) algorithms is to mined appropriate information from huge amounts of data while protecting at the same time thoughtful information[1,2]. The main goals a PPDM algorithm is:

1. A PPDM algorithm should have to thwart the discovery of sensible information. 2. It should be resistant to the various data mining techniques.
3. It should not compromise the access and the use of nonsensitive data.
4. It should not have an exponential computational complexity.

Many secure protocols have been proposed so far for data mining and machine learning techniques for decision tree classification, clustering, association rule mining, Neural Networks, Bayesian Networks. The main concern of these algorithms is to preserve the privacy of parties' sensitive data, while they gain useful knowledge from the whole dataset[3].

Most of the privacy-preserving data mining techniques apply a transformation which reduces the usefulness of the underlying data when it is applied to data mining techniques or algorithms. Privacy concerns can avoid building of centralized warehouse – in scattered among several places, no one are allowed to transfer their data to other place. In preserving privacy of data, the problem is how securely results are gained but not with data mining result but. As a simple example, suppose some hospitals want to get useful aggregated knowledge about a specific diagnosis from their patients' records while each hospital is not allowed, due to the privacy acts, to disclose individuals' private data. Therefore, they need to run a joint and secure protocol on their distributed database to reach to the desired information[1,3].

II. PPDM TECHNIQUES

Privacy preserving data mining received substantial attention and many researchers performed a good number of studies in the area. privacy preserving data mining has gained increasing popularity in data mining research community. PPDM has become an important issue in data mining research As a outcome, a whole new set of approaches were

ISSN: 2350-0328
International Journal of AdvancedResearch in Science, Engineering and Technology
Vol. 5, Issue 9 , September 2018

presented to allow mining of data, while at the same time leaving out the releasing any secretive and sensitive information. The majority of the existing approaches can be classified into two broad categories :

(i) Methodologies that protect the sensitive data itself in the mining process,
(ii) Methodologies that protect the sensitive data mining results (i.e. extracted knowledge) that were produced by the application of the data mining.

The first category refers to the methodologies that apply perturbation, sampling, generalization or suppression, transformation, etc. techniques to the original datasets in order to generate their sanitized counterparts that can be safely disclosed to untrustworthy parties. The goal of this category of approaches is to enable the data miner to get accurate data mining results when it is not provided with the real data. Secure Multiparty Computation methodologies that have been proposed to enable a number of data holders to collectively mine their data without having to reveal their datasets to each other[4].

The second category deals with techniques that prohibits the disclosure sensitive knowledge patterns derived through the application of data mining algorithms as well as techniques for downgrading the effectiveness of classifiers in classification tasks, such that they do not reveal sensitive information.

In difference to the centralized model, the Distributed Data Mining (DDM) model accepts that the individual's information is distributed across multiple places. Algorithms are developed within this area for the problem of efficiently receiving the mining results from all the data through these distributed sources. A simple method to data mining over multiple sources that will not share data is to run existing data mining tools at each place independently and combine the results .

PPDM tends to transform the original data so that the result of data mining task should not defy privacy constraints. Following is the list of dimensions on the basis of which different PPDM Techniques can be classified[5,6]:

1. Heuristic-based techniques: It is an adaptive modification that modifies only selected values that minimize the effectiveness loss rather than all available values. 2. Cryptography-based techniques: This technique includes secure multiparty computation where a computation is secure if at the completion of the computation, no one can know anything except its own input and the results. Cryptography based algorithms are considered for protective privacy in a distributed situation by using encryption techniques.

3. Reconstruction-based techniques: where the original distribution of the data is reassembled from the randomized data.

Based on these dimensions, different PPDM techniques may be classified into following five categories

A. Anonymization based PPDM
B. Perturbation based PPDM
C. Randomized Response based PPDM
D. Condensation approach based PPDM
E. Cryptography based PPDM

A) Anonymization based PPDM

The basic form of the data in a table consists of following four types of attributes:

(i) Explicit Identifiers is a set of attributes containing information that identifies a record owner explicitly such as name, SS number etc.
(i) Quasi Identifiers is a set of attributes that could potentially identify a record owner when combined with publicly available data.
(ii) Sensitive Attributes is a set of attributes that contains sensitive person specific information such as disease, salary etc.
(iii) Non-Sensitive Attributes is a set of attributes that creates no problem if revealed even to untrustworthy parties.

Anonymization refers to an approach where identity or/and sensitive data about record owners are to be hidden. It even assumes that sensitive data should be retained for analysis. It's obvious that explicit identifiers should be removed but still there is a danger of privacy intrusion when quasi identifiers are linked to publicly available data. Such attacks are called as linking attacks. For example attributes such as DOB, Sex, Race, and Zip are available in public records such as voter list. Such records are available in medical records also, when linked, can be used to infer the identity of the corresponding individual with high probability as shown in fig.3. Figure(1) shows k-anonymous with respect to quasi-identifier attributes[1,2,7].

ISSN: 2350-0328
International Journal of AdvancedResearch in Science, Engineering and Technology
Vol. 5, Issue 9 , September 2018

Sensitive data in medical record is disease or even medication prescribed. The quasi-identifiers like DOB, Sex, Race, Zip etc. are available in medical records and also in voter list that is publicly available. The explicit identifiers like Name, SS number etc. have been removed from the medical records. Still, identity of individual can be predicted with higher probability.

Replacing a value with less specific but semantically consistent value is called as generalization and suppression involves blocking the values. Releasing such data for mining reduces the risk of identification when combined with publically available data. But, at the same time, accuracy of the applications on the transformed data is reduced.

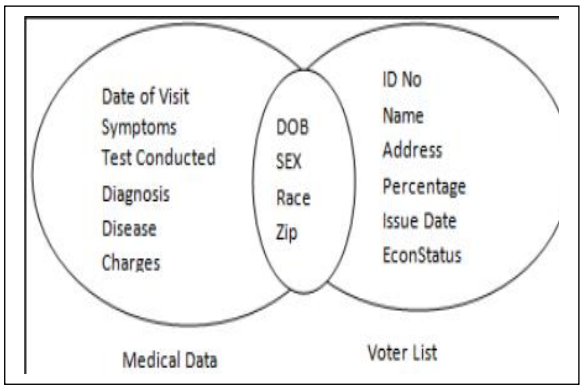


Figure (1) Linking Attack

B) Perturbation Based PPDM

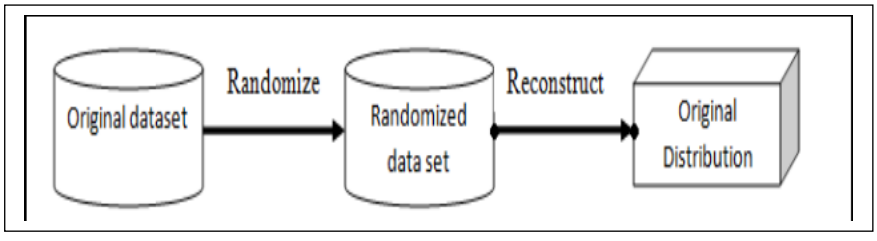
Data Perturbation is a technique for modifying data using random process. This technique apparently distorts sensitive data values by changing them by adding, subtracting or any other mathematical formula. This technique can handle different data types: character type, Boolean type, classification type and integer. In discrete data, it is required to pre-process the original data set. The pre-processing of data is classified into attribute coding and obtaining sets coded data set. The technique does not reconstruct the original data values, it only reconstructs the distribution.

Data distortion or data noise are different names for data perturbation. It is very important and critical to secure the sensitive data and data perturbation plays an important role in preserving the sensitive data. Distortion is done by applying different methods such as adding noise, data transpose matrix, by adding unknown values etc. In some perturbation approaches it is very difficult to preserve the original data[2,8] .

C) Randomized Response Based PPDM

Randomization is one of the promising approaches in privacy-preserving data mining. With this approach, original data are first disguised before being released to data collectors. Randomization-based disguise protects the private information of individuals, while still allowing the aggregated information (e.g., data distribution and patterns) to be preserved with reasonable accuracy.

The data collection process in randomization method is carried out using two steps. During first step, the data providers randomize their data and transfer the randomized data to the data receiver[2,9]. In second step, the data receiver rebuilds the original distribution of the data by using a distribution reconstruction algorithm The randomization response model is shown in figure(2)



Figure(2) Randomization Response Model

ISSN: 2350-0328
International Journal of AdvancedResearch in Science, Engineering and Technology
Vol. 5, Issue 9 , September 2018

D) Condensation approach based PPDM

This approach uses a methodology which condenses the data into multiple groups of pre-defined size. For each group, a certain level of statistical information about different records is maintained. This statistical information suffices to preserve statistical information about the mean and correlations across the different dimensions. Within a group, it is not possible to distinguish different records from one another. Each group has a certain minimum size k, which is referred to as the indistinguishability level of that privacy preserving approach. The greater the indistinguishability level, the greater the amount of privacy. At the same time, a greater amount of information is lost because of the condensation of a larger number of records into a single statistical group entity[10].

E) Cryptography Based PPDM

Consider a scenario where multiple medical institutions wish to conduct a joint research for some mutual benefits without revealing unnecessary information. In this scenario, research regarding symptoms, diagnosis and medication based on various parameters is to be conducted and at the same time privacy of the individuals is to be protected.
Such scenarios are referred to as distributed computing scenarios. The parties involved in mining of such tasks can be mutual untrusted parties, competitors; therefore protecting privacy becomes a major concern. Cryptographic techniques are ideally meant for such scenarios where multiple parties collaborate to compute results or share non sensitive mining results[11]

III. PRIVACY-PRESERVATION METHODS

Privacy-preservation methods can generally be executed at different steps of the data mining process:

1) **Privacy preserving data mining during Data Collection**
2) **)Privacy-Preserving Data Publishing**
3) **Distributed Privacy**
4) **Data Mining Output Privacy**

1) Privacy preserving data mining during Data Collection

In order to ensure effective data collection, it is important to design methods which can mine the data with a guarantee of privacy.
a) Approaches based on the goal of privacy preservation
The approaches Classified in two categories: **Data modification and Data sanitization**.
Data sanitization approaches aim to hide the critical rules and patterns existed in dataset. However, the **Data modification** approaches are hiding critical data and aiming to acquire valid results of data mining while private data cannot be reached directly and precisely[1,12].

Data modification techniques in PPDM can be classified in two principle groups of **perturbation-based and anonymization-based techniques** according to how the protection of privacy.
Anonymization techniquesare preventing from recognizing the critical data's characters and identity to preserve the privacy while **perturbation approach modify a part of data or the whole dataset** by means of determined techniques .
The current techniques in perturbation approach are classified in two categories **based on how they perturb datasets and particular Properties that will be preserved in data:**

- **Value-based Perturbation and**
- **Multi- Dimensional Perturbation.**

In the **Value-based Perturbation** the purpose is to preserve statistical characteristics and columns distribution while **Multi-Dimensional Perturbation** aims to hold MultiDimensional information.

ISSN: 2350-0328

International Journal of AdvancedResearch in Science, Engineering and Technology

Vol. 5, Issue 9 , September 2018

b) Anonymization Techniques

It is the process of either encrypting or removing personally identifiable information from data sets, so that the people whom the data describe remain anonymous. The attributes used for identifying the personal details, called "**Quasi-Identifier" (QI)** and itsvalues become modified so that they no longer uniquely represent individuals.

In particular, a table is K-anonymous if the QI attributes values of each record are identical to those of at least k-1 other records indistinguishable with at least k-1 other[2,13].

c) Generalization and Suppression

Generalization: is the process of identifying the parts of a whole, as belonging to the whole, such as A and B, A is a "generalization" of B, and B is a special case of A, if and only if

- every instance of concept B is also an instance of concept A; and
- there are instances of concept A which are not instances of concept B [2,14].

Suppression: replacing some attribute values (or parts of attribute values) by a special symbol that indicates that the value has been suppressed (e.g., "*" or "Any").

d) Partitioning

+ Single Dimensional
  - For each Xi , define non-overlapping single dimensional intervals that covers D_{xi}

+ Strict Multi-Dimensional
  - Define non-overlapping multi-dimensional intervals that covers D_{x1}… D_{xd}

Greedy Partitioning Algorithm

```
Anonymize(partition)
  if (no allowable multidimensional cut for partition)
    return φ : partition → summary
  else
    dim ← choose_dimension()
    fs ← frequency_set(partition, dim)
    splitVal ← find_median(fs)
    lhs ← {t ∈ partition : t.dim ≤ splitVal}
    rhs ← {t ∈ partition : t.dim > splitVal}
    return Anonymize(rhs) ∪ Anonymize(lhs)
```

Algorithm Example

- k = 2 , Dimension determined heuristically (Quasi-identifiers Zipcode , Age

Age	Sex	Zipcode	Disease
25	Male	53711	Flu
25	Female	53712	Hepatitis
26	Male	53711	Brochitis
27	Male	53710	Broken Arm
27	Female	53712	AIDS
28	Male	53711	Hang Nail

Patient Data

Age	Sex	Zipcode	Disease
[25-26]	Male	53711	Flu
[25-27]	Female	53712	Hepatitis
[25-26]	Male	53711	Brochitis
[27-28]	Male	[53710-53711]	Broken Arm
[25-27]	Female	53712	AIDS
[27-28]	Male	[53710-53711]	Hang Nail

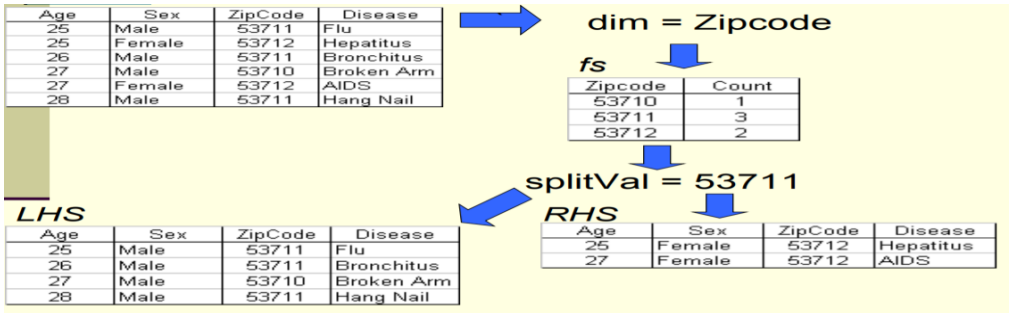
Anonymized Data

Iteration # 1 (full table) partition

ISSN: 2350-0328

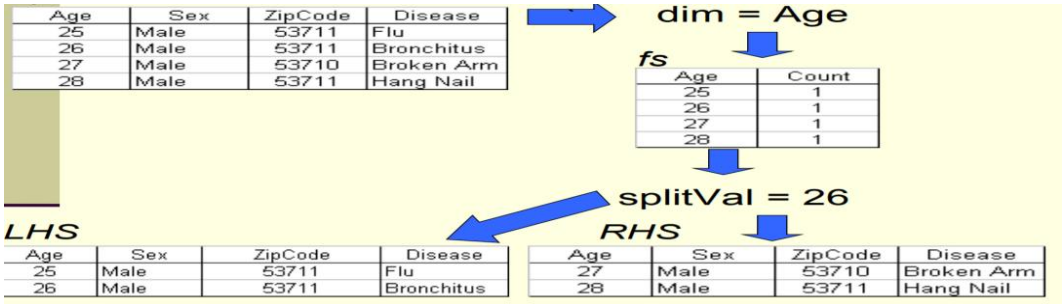
International Journal of AdvancedResearch in Science, Engineering and Technology

Vol. 5, Issue 9 , September 2018



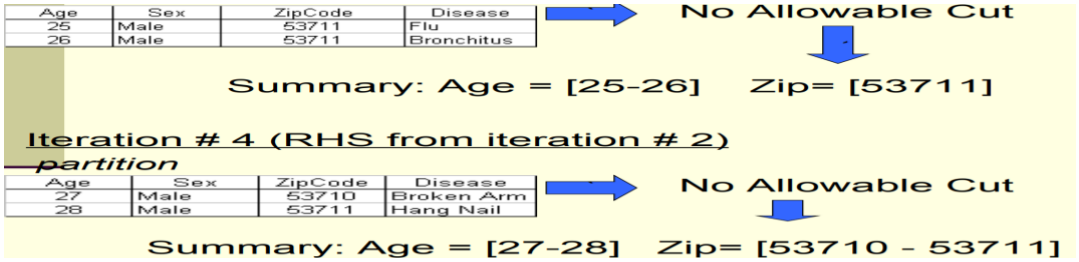
Iteration # 2 (LHS from iteration # 1)

Partition

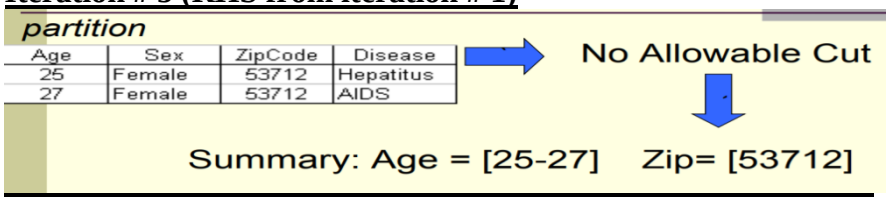


Iteration # 3 (LHS from iteration # 2)

Partition



Iteration # 5 (RHS from iteration # 1)



e) Value-based Perturbation Techniques

The main idea of this approach is to add random noise to the data values. Since the perturbing distribution is known, they can reach data mining goals by reconstructing their required aggregate distributions. The random perturbations are added to the data using a publicly available distribution ex: the uniform and the Gaussian distributions[15].

1- Rakesh's algorithm Distribution reconstruction

- Find distribution P(X|X+Y)
- three key points to understand it

- Bayes rule:
  P(X|X+Y) = P(X+Y|X) P(X)/P(X+Y)
- Conditional prob
- fx+y(X+Y=w|X=x) = fy(w-x)

ISSN: 2350-0328

International Journal of AdvancedResearch in Science, Engineering and Technology

Vol. 5, Issue 9 , September 2018

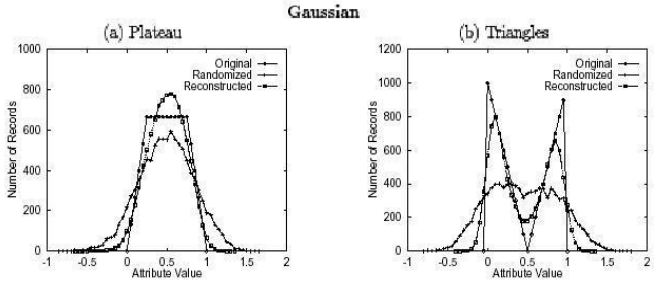
- Prob at the point *a* uses the average of all sampleestimates[1,16]

$$f'_X(a) = \frac{1}{n}\sum_{i=1}^{n} \frac{f_Y(w_i - a)\, f_X(a)}{\int_{-\infty}^{\infty} f_Y(w_i - z)\, f_X(z)\, dz}$$

- Stop criterion: the difference between two consecutive fx estimates is small

(1) f_X^0 := Uniform distribution
(2) $j := 0$ // Iteration number
repeat
(3) $f_X^{j+1}(a) := \frac{1}{n}\sum_{i=1}^{n} \frac{f_Y(w_i-a)\, f_X^j(a)}{\int_{-\infty}^{\infty} f_Y(w_i-z)\, f_X^j(z)dz}$
(4) $j := j + 1$
until (stopping criterion met)

- **Perturb a Discrete-Valued Attribute**



The basic idea is to replace the value of each data record under the attribute by another value that is chosen randomly from the attribute domain according to a probabilistic model. The general algorithm is explained in the following[17].

Algorithm : Random Substitution Perturbation RSP)

Input: a dataset $\mathcal{O}$ of n records, an attribute A with domain $\mathcal{U} = \{u_1, \ldots, u_N\}$, and a perturbation matrix $\mathbf{M}$ for $\mathcal{U}$.
Output: the perturbed dataset $\mathcal{P}$.
Method:

$\mathcal{P} = \emptyset$;
for each $o \in \mathcal{O}$ begin
1. $k = getIndex(o[A])$;
2. $p = o$;
3. obtain a random number r from a uniform distribution over $(0, 1]$;
4. find an integer $1 \leq h \leq N$ such that $\sum_{i=1}^{h-1} m_{i,k} < r \leq \sum_{i=1}^{h} m_{i,k}$;
5. set $p[A] = getValue(h)$;
6. add p to $\mathcal{P}$;

return $\mathcal{P}$;

2) Privacy-Preserving Data Publishing

Privacy-preserving data publishing is distinct from privacy-preserving data collection, because it is assumed that all the records are already available to a trusted party, who might be the current owner of the data. This party then wants to

ISSN: 2350-0328
International Journal of AdvancedResearch in Science, Engineering and Technology
Vol. 5, Issue 9 , September 2018

release (or publish) this data for analysis. For example, a hospital might wish to release anonymized records about patients to study the effectiveness of various treatment alternatives.
This form of data release is quite useful, because virtually any data mining algorithm can be used on the released data. To determine sensitive information about an individual, there are two main pieces of information that an attacker (or adversary) must possess.
1. Who does this data record pertain to? While a straightforward way to determine the identity is to use the identifying attribute (e.g., Social Security Number), such attributes are usually stripped from the data before release. As will be discussed later, these straightforward methods of sanitization are usually not sufficient, because attackers may use other attributes, such as the age and ZIP code, to make linkage attacks.
2. In addition to identifying attributes, data records also contain sensitive attributes that most individuals do not wish to share with others. For example, when a hospital releases medical data, the records might contain sensitive disease-related attributes. Different attributes in a data set may play different roles in either facilitating identification or facilitating sensitive information release[18,19].

There are three main types of attributes:

a. Explicit identifiers: These are attributes that explicitly identify an individual. For example, the Social Security Number (SSN) of an individual can be considered an explicit identifier. Because this attribute is almost always removed in the data sanitization process, it is not relevant to the study of privacy algorithms[1,20]
b. Pseudo-identifier or quasi-identifier (QID): These are attributes that do not explicitly identify an individual in isolation, but can nevertheless be used in combination to identify an individual by joining them with publicly available information, such as voter registration rolls. This kind of attack is referred to as a linkage attack. Examples of such attributes include the Age and ZIP code. Strictly speaking, quasi-identifiers refer to the specific combination of attributes used to make a linkage attack, rather than the individual attributes[2,21].
c. Sensitive attribute: These are attributes that are considered private by most individuals. For example, in a medical data set, individuals would not like information about their diseases to be known publicly. In fact, many laws in the USA, such as the Health Insurance Portability and Accountability Act (HIPAA), explicitly forbid the release of such information, especially when the sensitive attributes can be linked back to specific individuals.
Most of the discussion will be restricted to quasi-identifiers and sensitive attributes. To illustrate the significance of these attribute types, an example will be used[21].
Example
In table (1) ,The medical records of a set of individuals are illustrated. The SSN attribute is an explicit identifier that can be utilized to identify an individual directly. Such directly identifying information will almost always be removed from a data set before release. However, the impact of attributes such as the age and the ZIP code on identification is quite significant. While these attributes do not directly identify an individual, they provide very useful hints, when combined with other publicly available information.
For example, it is possible for a small geographic region, such as a ZIP code, to contain only one individual of a specific gender, race, and date of birth. When combined with publicly available voter registration rolls, one might be able to identify an individual from these attributes. Such a combination of publicly available attributes is referred to as a quasi-identifier[22].

Table(1) example of data table

SSN	Age	ZIP Code	Disease
012-345-6789	24	10598	HIV
823-627-9231	37	90210	Hepatitis C
987-654-3210	26	10547	HIV
382-827-8264	38	90345	Hepatitis C
847-872-7276	36	89119	Diabetes
422-061-0089	25	02139	HIV

To understand the power of quasi-identifiers, consider a snapshot of the voter registration rolls illustrated in Table (2). Even in cases, where the SSN is removed from Table (1) before release, it is possible to join the two tables with the use

ISSN: 2350-0328
International Journal of AdvancedResearch in Science, Engineering and Technology
Vol. 5, Issue 9 , September 2018

of the age and ZIP code attributes. This will provide a list of the possible matches for each data record. For example, Joy and Sue are the only two individuals in the voter registration rolls matching an individual with HIV in the medical release of Table (1). Therefore, one can tell with 50 % certainty that Joy and Sue have HIV. This is not desirable especially when an adversary has other background medical information about Joy or Sue to further narrow down the possibilities.

Similarly, William is the only individual in the voter registration rolls, who matches an individual with hepatitis C in the medical release. In cases, where only one data record in the voter registration rolls matches the particular combination of age and ZIP code, sensitive medical conditions about that individual may be fully compromised. This approach is referred to as a linkage attack. Most anonymization algorithms focus on preventing identity disclosure, rather than explicitly hiding the sensitive attributes. Thus, only the attributes which can be combined to construct quasi-identifiers are changed or specified approximately in the data release, whereas sensitive attributes are released in their exact form[23,3].

Name	Age	ZIP Code
Mary A.	38	90345
John S.	36	89119
Ann L.	31	02139
Jack M.	57	10562
Joy M.	26	10547
Victor B.	46	90345
Peter P.	25	02139
Diana X.	24	10598
William W.	37	90210
Sue G.	26	10547

Table (2) Example of a snapshot of fictitious voter registration

Many privacy-preserving data publishing algorithms assume that the quasi-identifiers are drawn out of a set of attributes that are not sensitive, because they can only be used by an adversary by performing joins with (nonsensitive) publicly available information. This assumption may, however, not always be reasonable, when an adversary has (sensitive) background information about a target at hand. Adversaries are often familiar with their targets, and they can be assumed to have background knowledge about at least a subset of the sensitive attributes.

In a medical application with multiple disease attributes, knowledge about a subset of these attributes may reveal the identity of the subject of the record. Similarly, in a movie collaborative filtering application, where anonymized ratings are released, it may be possible to obtain information about a particular user's ratings on a subset of movies, through personal interaction or other rating sources. If this combination is unique to the individual, then the other ratings of the individual are compromised as well. Thus, sensitive attributes also need to be perturbed, when background knowledge is available. Much of the work in the privacy literature assumes a rigid distinction between the role of publicly available attributes (from which the quasi-identifiers are constructed) and that of the sensitive attributes.

In other words, sensitive attributes are not perturbed because it is assumed that revealing them does not incur the risk of a linkage attack with publicly available information. There are, however, a few algorithms that do not make this distinction. Such algorithms generally provide better privacy protection in the presence of background information[24,1]

- ***The k-Anonymity Model***

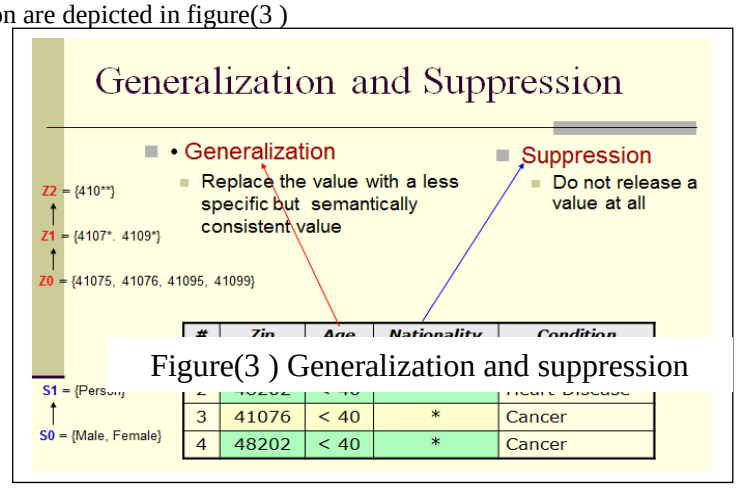
The k-anonymity model is one of the oldest ones for data anonymization, and it is credited with the understanding of the concept of quasi-identifiers and their impact on data privacy. The basic idea in k-anonymization methods is to allow release of the sensitive attributes, while distorting only the attributes which are available through public sources of information. Thus, even though the sensitive attributes have been released, they cannot be linked to an individual

ISSN: 2350-0328
International Journal of AdvancedResearch in Science, Engineering and Technology
Vol. 5, Issue 9 , September 2018

through publicly available records. Before discussing the anonymization algorithms, some of the most common techniques for data distortion will be discussed.

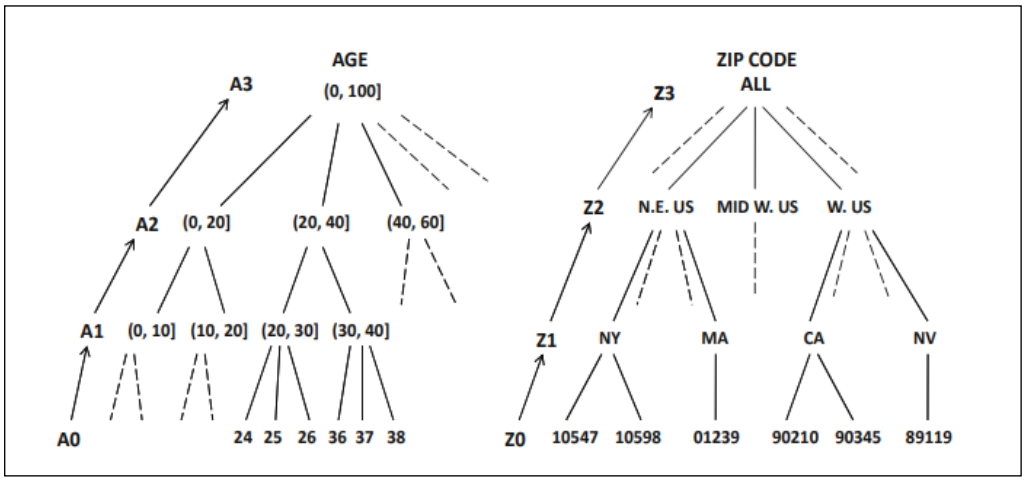
a. **Suppression**: In this approach, some of the attribute values are suppressed. Depending on the algorithm used, the suppression can be done in a variety of ways. For example, one might omit some of the age or ZIP code attribute values from a few selected data records in Table (1). Alternatively, one might completely omit the entire record for a specific individual (row suppression) or the age attribute from all individuals (column suppression). Row suppression is often utilized to remove outlier records because such records are difficult to anonymize. Column suppression is commonly used to remove highly identifying attributes, or explicit identifiers, such as the SSN[25].

b. **Generalization:** In the case of generalization, the attributes are specified approximately in terms of a particular range. For example, instead of specifying Age = 26 and Location (ZIP Code) = 10547 for one of the entries of Table (1), one might generalize it to Age ∈ [25, 30] and Location (State) = New York. By specifying the attributes approximately, it becomes more difficult for an adversary to perform linkage attacks. While numeric data can be generalized to specific ranges, the generalization of categorical data is somewhat more complicated. Typically, a generalization hierarchy of the categorical attribute values needs to be provided, for use in the anonymizationprocess[26]. Examples of suppression and generalization are depicted in figure(3)



Figure(3) Generalization and suppression

For example, a ZIP code may be generalized to a city, which in turn may be generalized to a state, and so on. There is no unique way of specifying a domain hierarchy. Typically, it needs to be semantically meaningful, and it is specified by a domain expert as a part of the input to the anonymization process. An example of a generalization taxonomy of categorical attributes for the location attribute of Table (1) is provided in Fig. (3). This hierarchy of attribute values has a tree structure, and is referred to as a value generalization hierarchy. The notations A0 ...A3 and Z0 ...Z4 in Fig. (3) denote the domain generalizations at different levels of granularity. The corresponding domain generalization hierarchies are also illustrated in the Fig. (3) by the single path between Z0 ...Z4 and A0 ...A4.



ISSN: 2350-0328

International Journal of AdvancedResearch in Science, Engineering and Technology

Vol. 5, Issue 9 , September 2018

Definition (k-anonymity)

A data set is said to be k-anonymized, if the attributes of each record in the anonymized data set cannot be distinguished from at least (k −1) other data records.

This group of indistinguishable data records is also referred to as an equivalence class. To understand how generalization and suppression can be used for anonymization, consider the data set in Table (1).

An example of a 3-anonymized version of this table is illustrated in Table (3). The SSN has been fully suppressed with column-wise suppression and replaced with an anonymized row index. Such explicit identifiers are almost always fully suppressed in anonymization. The two publicly available attributes corresponding to the age and ZIP code are now generalized and specified approximately. The subjects of the row indices 1, 3, and 6 can no longer be distinguished by using linkage attacks because their publicly available attributes are identical. Similarly, the publicly available attributes of row indices 2, 4, and 5 are identical. Thus, this table contains two equivalence classes containing three records each, and the data records cannot be distinguished from one another within these equivalence classes. In other words, anadversary can no longer match the identification of individual data records with voter registration rolls exactly[3,10].

Table (3)Example of a 3-anonymized version of Table 2

Row Index	Age	ZIP Code	Disease
1	[20, 30]	Northeastern US	HIV
2	[30, 40]	Western US	Hepatitis C
3	[20, 30]	Northeastern US	HIV
4	[30, 40]	Western US	Hepatitis C
5	[30, 40]	Western US	Diabetes
6	[20, 30]	Northeastern US	HIV

If any matching is found, then it is guaranteed that at least k = 3 records in the data set will match any particular individual in the voter registration roll. The ZIP code is generalized with the use of the prespecified value generalization hierarchy of Fig. (3). The generation of a domain generalization hierarchy for a categorical attribute can be done in several ways, and depends on the skill of the analyst responsible for the privacy modifications. An alternate example of a domain generalization hierarchy for the ZIP code attribute is illustrated in Fig.(4). A value generalization hierarchy on the continuous attributes does not require any special domain knowledge because it can be directly created by the analyst, using the actual distribution of the continuous values in the underlying data[27,3]. This requires a simple hierarchical discretization of the continuous attributes

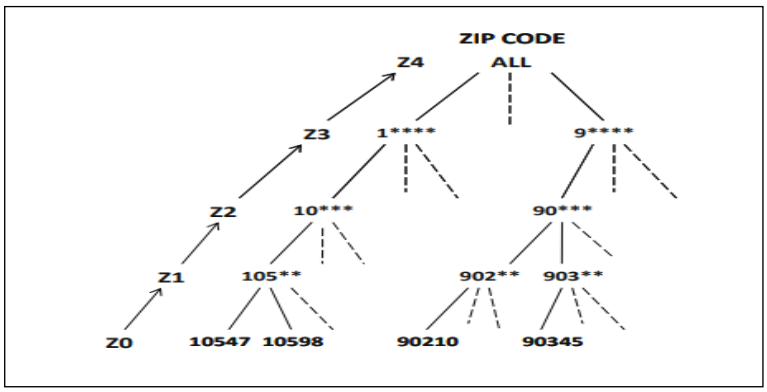


Figure (4) An alternate value- and corresponding domain-generalization hierarchy for the ZIP code attribute

The goal of the privacy-preservation algorithms is to replace the original values in the data (numeric or discrete), with one of the discrete values illustrated in the taxonomy trees of Fig. (3). Thus, the data is recoded in terms of a new set of

ISSN: 2350-0328
International Journal of AdvancedResearch in Science, Engineering and Technology
Vol. 5, Issue 9 , September 2018

discrete values. In most cases, the numeric attributes do retain their ordering, because the corresponding ranges are ordered. Different algorithms use different rules in the recoding process. These different ways of recoding attributes may be distinguished as follows:

- **Global versus local recoding:** In global recoding, a given attribute value is always replaced with the same discrete counterpart from the domain generalization hierarchy over all data records. Consider the aforementioned example of Fig. (3), in which ZIP code can be generalized either to state or region. In global recoding, the particular ZIP code value of 10547 needs to be consistently replaced by either Northeastern US, or New York over all the data records. However, for a different ZIP code such as 90210, a different level of hierarchy may be selected than for the 10547 value, as long as it is done consistently for a particular data value (e.g., 10547 or 90210) across all data records. In local recoding, different data records may use different generalizations for the same data value. For example, one data record might use Northeastern US, whereas another data record might use New York for 10547. While local recoding might seem to be better optimized, because of its greater flexibility, it does lose a different kind of information. In particular, because the same ZIP code might map to different values, such as New York and Northeastern US, the similarity computation between the resulting data records may be less accurate. Most of the current privacy schemes use global recoding[28].
- **Full-domain generalization:** Full-domain generalization is a special case of global recoding. In this approach, all values of a particular attribute are generalized to the same level of the taxonomy. For example, a ZIP code might be generalized to its state for all instances of the attribute. In other words, if the ZIP code 10547 is generalized to New York, then the ZIP code 90210 must be generalized to California. The various hierarchical alternatives for full-domain generalization of the age attribute are denoted by A0, A1, A2, and A3 in Fig. (3). The possible full-domain generalization levels of the ZIP code are denoted by Z0, Z1, Z2, and Z3. In this case, Z3 represents the highest level of generalization (column suppression), and Z0 represents the original values of the ZIP code attribute. Thus, once it is decided that the anonymization algorithm should use Z2 for the ZIP code attribute, then every instance of the ZIP code attribute (Z0) in the data set is replaced with its generalized value in Z2. This is the reason that the approach is referred to as full-domain generalization, as the entire domain of data values for a particular attribute is generalized to the same level of the hierarchy. Full-domain generalization is the most common approach used in privacy-preserving data publishing[28].

3) Distributed Privacy

In distributed privacy-preserving data mining, the goal is to mine shared insights across multiple participants owning different portions of the data, without compromising the privacy of local statistics or data records. The key is to understand that the different participants may be partially or fully adversaries/competitors, and may not wish to provide full access of their local data and statistics to one another. However, they might find it mutually beneficial to extract global insights over all the data owned by them. The data may be partitioned either horizontally or vertically across the different participants.

In horizontal partitioning, the data records owned by different adversaries have the same attributes, but different adversaries own different portions of the database. For example, a set of supermarket chains may own similar data related to customer buying behavior, but the different stores may show somewhat different patterns in their transactions because of factors specific to their particular business.

In vertical partitioning, the different sites may contain different attributes for the same individual. For example, consider a scenario in which a database contains transactions by various customers. A particular customer may buy different kinds of items at stores containing complementary products such as jewelery, apparel, cosmetics, etc. In such cases, the aggregate association analysis across different participants can provide insights, that cannot be inferred from any particular database[28,1,2]. Examples of horizontal and vertically partitioned data are provided in Figure(5)

ISSN: 2350-0328
International Journal of AdvancedResearch in Science, Engineering and Technology
Vol. 5, Issue 9 , September 2018

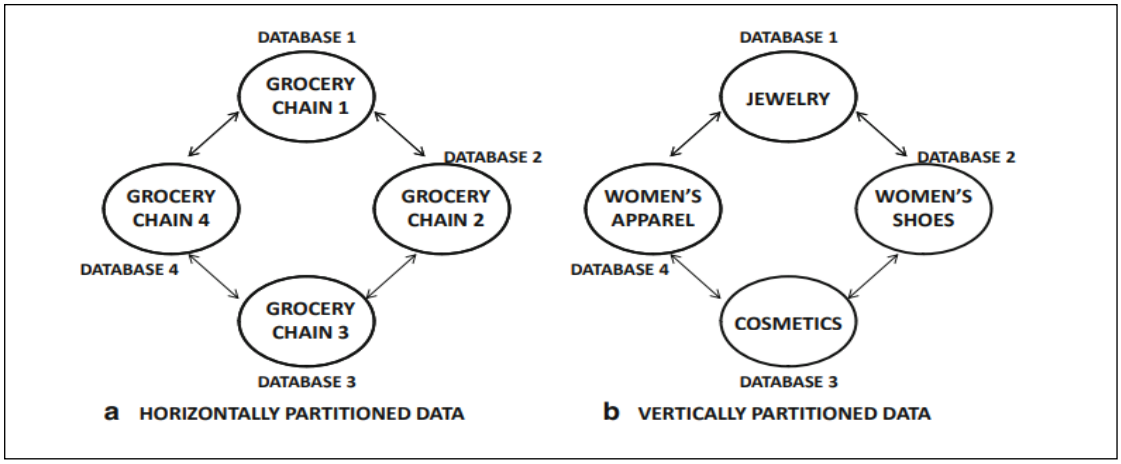


Figure (5) horizantly and vertically partitioned data

At the most primitive level, the problem of distributed privacy-preserving data mining overlaps closely with a field in cryptography for determining secure multi-party computations. In this field, functions are computed over inputs provided by multiple recipients without actually sharing the inputs with one another. For example, in a two-party setting, Alice and Bob may have two inputs x and y, respectively, and may wish to compute the function f(x, y) without revealing x or y to each other. This problem can also be generalized across k parties for computing the k argument function h(x1 ...xk).

Many data mining algorithms may be viewed in the context of repetitive computations of primitive functions such as the scalar dot product, secure sum, secure set union, etc. For example, the scalar dot product of the binary representation of an itemset and a transaction can be used to determine whether or not that itemset is supported by that transaction. Similarly, scalar dot products can be used for similarity computations in clustering. To compute the function f(x, y) or h(x1 ...,xk), a protocol needs to be designed for exchanging information in such a way that the function is computed without compromising privacy[1,29].

- **Secure Multiparty Computation**

The basic idea of Secure Multiparty Computation is that a computation is secure if at the end of the computation, no party knows anything except its own input and the results. One way to view this is to imagine a trusted third party – everyone gives their input to the trusted party, who performs the computation and sends the results to the participants. Now imagine that we can achieve the same result without having a trusted party.

Obviously, some communication between the parties is required for any interesting computation – how do we ensure that this communication doesn't disclose anything? The answer is to allow non-determinism in the exact values sent in the intermediate communication (e.g., encrypt with a randomly chosen key), and demonstrate that a party with just its own input and the result can generate a "predicted" intermediate computation that is as likely as the actual values. This has been shown possible, however the general method given does not scale well to data mining sized problems. A detailed discussion of Secure Multiparty Computation is given elsewhere in this issue, and we encourage readers who want a deep understanding of the following material to start with that article. We now give some examples of privacy preserving computations, show some of the subtleties involved in ensuring that such a computation is truly secure[1,2,10].

1) Techniques

a) Secure Sum

Secure sum is often given as a simple example of secure multiparty computation. Distributed data mining algorithms frequently calculate the sum of values from individual sites. Assuming three or more parties and no collusion, the following method securely computes such a sum.

Assume that the value

$$v = \sum_{l=1}^{s}$$

ISSN: 2350-0328
International Journal of AdvancedResearch in Science, Engineering and Technology
Vol. 5, Issue 9 , September 2018

to be computed is known to lie in the range [0..n]. One site is designated the master site, numbered 1. The remaining sites are numbered 2..s. Site 1 generates a random number R, uniformly chosen from [0..n]. Site 1 adds this to its local value v1, and sends the sum R + v1 mod n to site 2. Since the value R is chosen uniformly from [1..n], the number R + v1 mod n is also distributed uniformly across this region, so site 2 learns nothing about the actual value of v1[20]. See figure (6).

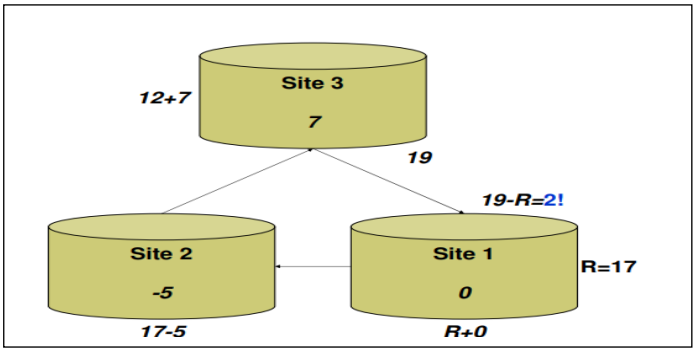


Figure (6) secure computation of a sum

$$V = R + \sum_{j=1}^{l-1} v_j \bmod n.$$

Since this value is uniformly distributed across [1..n], i learns nothing. Site i then computes

$$R + \sum_{j=1}^{l} v_j \bmod n = (v_j + V) \bmod n$$

and passes it to site l + 1. Site s performs the above step, and sends the result to site 1. Site 1, knowing R, can subtract R to get the actual result.
Note that site 1 can also determine
$\sum_{l=2}^{s} v_l$

by subtracting v1. This is possible from the global result regardless of how it is computed, so site 1 has not learned anything from the computation. Figure 1 depicts how this method operates. This method faces an obvious problem if sites collude. Sites l − 1 and l + 1 can compare the values they send/receive to determine the exact value for vl. The method can be extended to work for an honest majority. Each site divides vl into shares. The sum for each share is computed individually. However, the path used is permuted for each share, such that no site has the same neighbor twice. To compute vl , the neighbors of l from each iteration would have to collude. Varying the number of shares varies the number of dishonest (colluding) parties required to violate security[20,22].

b)Secure Set Union

Secure union methods are useful in data mining where each party needs to give rules, frequent itemsets, etc., without revealing the owner. The union of items can be evaluated using SMC methods if the domain of the items is small. Each party creates a binary vector where 1 in the i th entry represents that the party has the i th item. After this point, a simple circuit that or's the corresponding vectors can be built and it can be securely evaluated using general secure multi-party circuit evaluation protocols.

However, in data mining the domain of the items is usually large. To overcome this problem a simple approach based on commutative encryption is used. An encryption algorithm is commutative if given encryption keys K1, . . . ,Kn ∈ K, for any m in domain M, and for any permutation i, j, the following two equations hold:

ISSN: 2350-0328

International Journal of AdvancedResearch in Science, Engineering and Technology

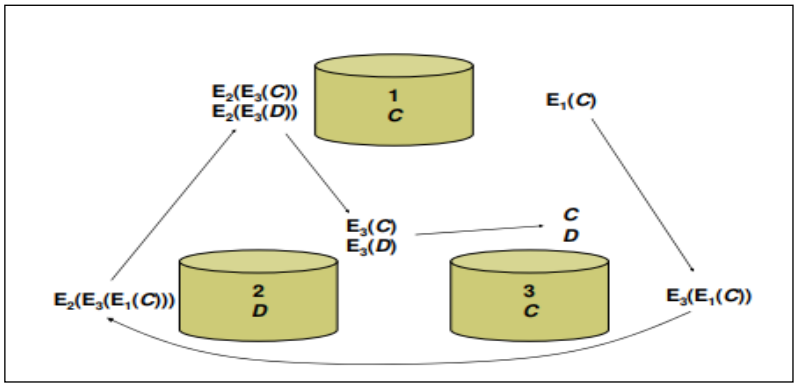
Vol. 5, Issue 9 , September 2018

$$E_{K_{i_1}}(\dots E_{K_{i_n}}(M)\dots) = E_{K_{j_1}}(\dots E_{K_{j_n}}(M)\dots) \qquad (1)$$

$$\forall M_1, M_2 \in M \text{ such that } M_1 \neq M_2 \text{ and for given } k,\ \epsilon < \frac{1}{2^k}$$

$$Pr(E_{K_{i_1}}(\dots E_{K_{i_n}}(M_1)\dots) = E_{K_{i_1}}(\dots E_{K_{i_n}}(M_2)\dots)) < \epsilon$$

The main idea is that each site encrypts its items. Each site then encrypts the items from other sites. Since equation 1 holds, duplicates in the original items will be duplicates in the encrypted items, and can be deleted[10,15]. (Due to equation 2, only the duplicates will be deleted.) In addition, the decryption can occur in any order, so by permuting the encrypted items we prevent sites from tracking the source of an item. The algorithm for evaluating the union of the items is given in the following algorithm, and an example is shown in Figure (7).



Figure(7) Determining the Union of a set of items

Algorithm 1 Finding secure union of items

Require: N is number of sites and Union_set = ∅ initially
{Encryption of all the rules by all sites}
for each site i **do**
  for each $X \in S_i$ **do**
    $M = newarray[N]$;
    $Xp = encrypt(X, e_i)$;
    $M[i] = 1$;
    Union_set ⋃ (Xp,M);
  end for
end for{Site i encrypts its items and adds them to the global set. Each site then encrypts the items it has not encrypted before}

for each site i **do**
  for each tuple (r,M) ∈ Union_set **do**
    if M[i] == 0 **then**
      rp=encrypt(r,e_i);
      M[i]=1;
      Mp= M ;
      Union_set=(Union_set-{(r,M)}) ⋃ {(rp,Mp)};
    end if
  end for
end for

for (r,M) ∈ Union_set and (rp,Mp) ∈ Union_set **do**
{check for duplicates}
  if r==rp **then**
    Union_set= Union_set-{(r,M)} {Eliminate duplicate items before decrypting};
  end if
end for

ISSN: 2350-0328

International Journal of AdvancedResearch in Science, Engineering and Technology

Vol. 5, Issue 9 , September 2018

Clearly the algorithm finds the union without revealing which item belongs to which site. It is not, however, secure under the definitions of secure multi-party computation. It reveals the number of items that exist commonly in two sites, e.g. if k sites have an item in common, there will be an (encrypted) item duplicated k times. This does not reveal which items these are, but a truly secure computation (as good as each site giving its input to a "trusted party") could not reveal even this count. Allowing innocuous information leakage (the number of items that is owned by two sites) allows an algorithm that is sufficiently secure with much lower cost than a fully secure approach[1,13,14].

We can prove that other than the size of intersections and the final result, nothing is revealed. By assuming that the count of duplicated items is part of the final result, a Secure Multiparty Computation proof is possible

c)Secure Size of Set Intersection

Consider several parties having their own sets of items from a common domain. The problem is to securely compute the cardinality/size of the intersection of these local sets. Formally, given k parties P1 . . . Pk having local sets S1 . . . Sk, we wish to securely compute |S1 ∩ · · · ∩ Sk|. We can do this is using a parametric commutative one way hash function. One way of getting such a hash function is to use commutative public key encryption, such as Pohlig Hellman, and throw away the decryption keys.

All k parties locally generate their public key-pair (Ei, Di) for a commutative encryption scheme. (They can throw away their decryption keys since these will never be used.) Each party encrypts its items with its key and passes it along to the other parties. On receiving a set of (encrypted) items, a party encrypts each item and permutes the order before sending it to the next party. This is repeated until every item has been encrypted by every party[10].

Since encryption is commutative, the resulting values from two different sets will be equal if and only if the original values were the same (i.e., the item was present in both sets). Thus, we need only count the number of values that are present in all of the encrypted itemsets. This can be done by any party. None of the parties is able to know which of the items are present in the intersection set because of the encryption. The complete protocol is shown in the following algorithm[20]

Algorithm 2 Securely computing size of intersection set

Require: k sites
Require: each site has a local set S_i
Generate the commutative encryption key-pair (E_i, D_i)
{Throw away the decryption keys, since they will not be needed.}
$M = S_i$
for $k-1$ steps **do**
  $M' = newarray[|M|]$
  j=0;
  for each $X \in M$ **do**
    $M'[j++] = encrypt(X, E_i)$
  end for
  permute the array M' in some random order
  send the array M' to site $i+1 \mod k$
  receive array M from site $i-1 \mod k$
end for
$M' = newarray[|M|]$
j=0;
for each $X \in M$ **do**
  $M'[j++] = encrypt(X, E_i)$
end for
permute the array M' in some random order
send M' to site $i \mod 2$ {This prevents a site from seeing it's own encrypted items}
sites 0 and 1 produce array I_0 and I_1 containing only (encrypted) items present in all arrays received.
site 1 sends I_1 to site 0
site 0 broadcasts the result $|I_0 \cup I_1|$

d)Scalar Product

Scalar product is a powerful component technique. Many data mining problems can essentially be reduced to computing the scalar product.

ISSN: 2350-0328

International Journal of AdvancedResearch in Science, Engineering and Technology

Vol. 5, Issue 9 , September 2018

The problem can be formally defined as follows: Assume 2 parties P1 and P2 each have a vector of cardinality n; i.e. P1 has X~ = (x1 . . . xn) and P2 has Y~ = (y1 . . . yn). The problem is to securely compute the scalar product of the two vectors, i.e.,. $\sum_{i=1}^{n} x_i * y_i$

Recently, there has been a lot of research into this problem, which has given rise to many different solutions with varying degrees of accuracy, communication cost and security.

Note that all of these techniques are limited to the 2-party version of the problem and cannot easily be extended to the general case. In the problem is modeled as Secure Multiparty Computation and the present a solution using cryptographic techniques (oblivious transfer). This, however, is not very efficient. The key is to use linear combinations of random numbers to disguise vector elements and then do some computations to remove the effect of these randoms from the result. The solution is briefly explained in the following algorithm. Though this method does reveal more information than just the input and the result, it is efficient and suited for large data sizes, thus being useful for data mining[20,25]

Algorithm 3 Computing the scalar product

Require: N=2 is number of sites, site A and site B
Require: Each site has a vector of cardinality n. Thus,
Require: A has $\vec{X} = (x_1, \ldots, x_n)$ and
Require: B has $\vec{Y} = (y_1, \ldots, y_n)$
Both A and B together decide on a random $n \times n/2$ matrix C
for Site A **do**
A generates a random vector R of cardinality $n/2$ ($\vec{R} = R_1, \ldots, R_{n/2}$)
A generates the $n \times 1$ addition matrix X' by multiplying C with R (i.e. $X' = C \times R$)
A generates $X'' = X + X'$
A sends X'' to B
end for{First message from A to B}
for Site B **do**
B generates the scalar product S' of X'' and Y (i.e. $S' = \sum_{i=1}^{n} x_i'' * y_i$)
B also generates the $n \times 1$ matrix $Y' = C^T \times Y$
B sends both S' and Y' to A
end for{First message from B to A}
for Site A **do**
A generates the subtraction factor $S'' = \sum_{i=1}^{n} Y_i' * R_i$
A generates the required scalar product $S = S' - S''$
A reports the scalar product S to B
end for{Second message from A to B}

4) **Data Mining Output Privacy**

The output of data mining algorithms can be very informative to an adversary. For example, consider an association rule mining algorithm, in which the following rule is generated with high confidence:

(Age = 26, ZIP Code = 10562) ⇒ HIV

the discovery of this rule may result in the unforeseen disclosure of private information about an individual[10]l.

Below, a description of the most common techniques to preserve privacy to the output of the data mining is presented.

a) Association Rule Hiding

ISSN: 2350-0328

International Journal of AdvancedResearch in Science, Engineering and Technology

Vol. 5, Issue 9 , September 2018

Association rule hiding is a privacy-preserving technique whose **objective is to mine all non-sensitive rules, while no sensitive rule is discovered.** Association rule hiding algorithms prevents the sensitive rules from being revealed out.
Two broad approaches are used for association rule hiding:
Distortion: the entry for a given transaction is **modified to a different value**. Since, typically dealing with binary transactional data sets, the entry value is flipped.
Blocking: the **entry is not modified, but is left incomplete**. Thus, unknown entry values are used to prevent discovery of association rules[5,6].

- **Hiding Of Sensitive Association Rules Algorithm**

Representative rules (RR) is a set of rules that allow deducing all association rules without accessing a database. The process of generating representative rules is decomposed in to two sub processes: **frequent item-sets generations and generationof RR from frequent item-sets.** The cover operator was introduced for driving a set of association rules .

The cover of the rule $A \Rightarrow B , A \neq \emptyset$ is defined as follows:

$$C(A => B) = \{A \cup B => V \mid Z , V \subseteq B \text{ and } Z \cap V = \phi \text{ and } V \neq \phi\}$$

Each rule in $C\ (A \Rightarrow B)$consists of a subset of items occurring in the rule $A \Rightarrow B$.
Formally, a set of representative rules (RR) for a given association rules (AR) can be defined as follows:

$$RR= \{ r \in AR \mid \neg\exists\ r' \in AR, r\ ' \neq r \text{ and } r \in C(r')\}$$

Each rule in RR is called representative association rule. **No representative rule may belong in thecover of another association rule.**
The following algorithm steps for hiding the sensitive rules (**sensitive rules are those rules that contain sensitive item(s)).**

1- A Database, value of min_support, min_confidence, and a set of sensitive items are given as input this algorithm.
2- Association rules are generated using association rule mining algorithm.
3- All the rules containing sensitive item(s) either in the left or in the right are selected.
4- Rules are converted in representative rules (RRs) format.
5- A rule from the set of RR's, which has sensitive item on the left of the RR is selected.
6- The sensitive item(s) from the transaction that completely supports the RR is removed and add the same sensitive item to a transaction which partially supports RR.
7- The confidence of the rules in U is recomputed[1,2,3].

The following example illustrated for a given set of transactional data in Table -1

- For the Dataset given in Table 4 at a min_supp of 33% and a min_conf of 70 % and sensitive item H= {milk}
- choose all the rules containing 'butter' either in RHS or LHS and represent them in representative rule format. Out of the 8 association rules the rules containing sensitive items are 6 as shown in Table 5

Table4 -: Transactional Dataset1

TID	ITEMS
T1	bread, butter, milk
T2	bread, butter, milk, cheese
T3	butter, milk, fruits
T4	bread, milk, cheese, fruits
T5	cheese, fruits
T6	bread, butter

1

ISSN: 2350-0328

International Journal of AdvancedResearch in Science, Engineering and Technology

Vol. 5, Issue 9 , September 2018

- From this rules set select the rules that can be represented in the form of representative rules Like **milk=> butter and milk=>bread can be represented as milk=> bread, butter**

- Now delete milk from a transaction where bread, butter, milk all the three are present and add milk to a transaction where bread and butter both are absent or only one of them is present. **For this we change transaction T2 to bread, butter, cheese and transaction T5 to milk, cheese, and fruits.** This results in changing the position of the sensitive item without changing its support[1,10]. This is shown in Table 6

Table - 5 : Sensitive association rules (w.r.t sensitive item 'milk')

AR	SUPP	CONF
bread => milk	50	75
milk => bread	50	75
bread, cheese => butter	33.333	100
milk, cheese => bread	33.333	100
butter => milk	50	75
milk => butter	50	75

Table6-: Modified Dataset1

TID	ITEMS
T1	bread, butter, milk
T2	bread, butter, cheese
T3	butter, milk, fruits
T4	bread, milk, cheese, fruits
T5	milk, cheese, fruits
T6	bread, butter

Table- 7: Association rules remaining unhidden after modifying the Dataset1

AR	SUPP	CONF
butter=> bread	50	75
bread=>butter	50	75

i.e. all the rules of the original association rules set containing sensitive items on the LHS or on the RHS are hidden.

b) Downgrading Classifier Effectiveness -

Data is sanitized to reduce classifier' accuracy and consequently the possibility of inferring sensitive data. Since some rule based classifiers use association rule mining methods as subroutines, association rule hiding methods are also applied to downgrade the effectiveness of the classifier[4]**.**

c) Query Auditing and Inference Control

Techniques to prevent information disclosure from sequences of aggregate data queries .Two broad approaches are designed in order to reduce the likelihood of sensitive data discovery:

Query Auditing: In query auditing, deny one or more queries from a sequence of queries. The queries to be denied are chosen such that the sensitivity of the underlying data is preserved.

Query InferenceControl: In this case, perturb the underlying data or the query result itself. The perturbation is engineered in such a way, so as to preserve the privacy of the underlying data[5].

IV. CONCLUSION

The main objective of privacy preserving data mining is developing algorithm to hide or provide privacy to certain sensitive information so that they cannot be disclosed to unauthorized parties or intruder. Although a Privacy and accuracy in case of data mining is a pair of ambiguity. Succeeding one can lead to adverse effect on other. In this, we made an effort to review a good number of existing PPDM techniques. Finally, we conclude there does not exists a single privacy preserving data mining algorithm that outperforms all other algorithms on all possible criteria like performance, utility, cost, complexity, tolerance against data mining algorithms etc. Different algorithm may perform better than another on one particular criterion.

REFERENCES

1. Charu C. Aggarwal& Philip S. Yu ."Privacy-Preserving Data Mining Models And Algorithms " , Springer Science+Business Media, LLC. 2008.
2. Kasthuri S AndMeyyappan T. " Hiding Sensitive Association Rule Using Heuristic Approach ".International Journal Of Data Mining & Knowledge Management Process (IJDKP) Vol.3, No.1, January 2013.
3. K. Sathiyapriya And Dr. G. SudhaSadasivam ." A Survey On Privacy Preserving Association Rule Mining " . International Journal Of Data Mining & Knowledge Management Process (Ijdkp) Vol.3, No.2, March 2013 .

ISSN: 2350-0328

International Journal of AdvancedResearch in Science, Engineering and Technology

Vol. 5, Issue 9 , September 2018

4. Jim Dowd, ShouhuaiXu, and WeiningZhang " Privacy-Preserving Decision Tree Mining Based On Random Substitutions" . Springer-Verlag Berlin Heidelberg 2006 .
5. Charu C. Aggarwal" Data Mining: The Textbook" . Springer International Publishing Switzerland 2015.
6. RakeshAgrawal&RamakrishnanSrikant , " Privacy-Preserving Data Mining " . 2000 ACM.
7. Dhyanendra Jain, et.al , "Hiding Sensitive Association Rules without Altering the Support of Sensitive Item(s)",,International Journal of Artificial Intelligence & Applications (IJAIA), Vol.3, No.2, March 2012
8. Josenildo C. da Silva2 , Chris Giannella , RuchitaBhargava , HillolKargupta , and Matthias Klusch "Distributed Data Mining and Agents" .
9. Charu C. Aggarwal , "Data Mining The Textbook " , springer , IBM T.J. Watson Research Center Yorktown Heights New York USA , 2015
10. Charu C. Aggarwal and Philip S. Yu , "Privacy-Preserving Data Mining Models and Algorithms",Springer , USA , 20018
11. P. Cynthia Selvi and A.R.MohammedShanavas , "Output Privacy Protection With Pattern-Based Heuristic Algorithm" , International Journal of Computer Science & Information Technology (IJCSIT) Vol 6, No 2, April 2014
12. XinjingGe and Jianming Zhu , "Privacy Preserving Data Mining" , School of Information, Central University of Finance and Economics Beijing, China.
13. ZHUOJIA XU , "Analysis of Privacy Preserving Distributed Data Mining Protocols" , Master Thesis , School of Engineering and Science, Faculty of Health, Engineering and Science, VICTORIA UNIVERSITY
14. Mayur B Tank and Tushar A Champaneria , "Privacy Preserving Distributed Data Mining Techniques" , IJIRST –International Journal for Innovative Research in Science & Technology| Volume 1 | Issue 9 | February 2015 ISSN (online): 2349-6010.
15. NissimMatatov, LiorRokach and OdedMaimon , "Privacy-preserving data mining: A feature set partitioning approach", Information Sciences 180 (2010) 2696–2720.
16. FrenyPresswala ,AmitThakkar and , Nirav Bhatt , "International Journal of Innovative and Emerging Research in Engineering", International Journal of Innovative and Emerging Research in Engineering Volume 2, Issue 2, 2015.
17. Lijie Zhang and Weining Zhang , "Generalization-Based Privacy-Preserving Data Collection", Springer-Verlag Berlin Heidelberg 2008
18. Igor NaiFovino and Marcelo Masera , "State of the Art in Privacy Preserving Data Mining', 16th January 2008
19. HinaVaghashia and AmitGanatra , "A Survey: Privacy Preservation Techniques in Data Mining", International Journal of Computer Applications (0975 – 8887) Volume 119 – No.4, June 2015
20. Yehuda Lindell and Benny Pinkas , "Privacy Preserving Data Mining" .
21. Amar Paul Singh and Ms. DhanshriParihar , "A Review of Privacy Preserving Data Publishing Technique", International Journal of Emerging Research in Management &Technology ISSN: 2278-9359 (Volume-2, Issue-6) , June 2013.
22. S.Gokila and Dr.P.Venkateswari , "A SURVEY ON PRIVACY PRESERVING DATA PUBLISHING" , International Journal on Cybernetics & Informatics (IJCI) Vol. 3, No. 1, February 2014
23. Benjamin C. M. Fung , "PRIVACY-PRESERVING DATA PUBLISHING" , PhD Thesis , SIMON FRASER UNIVERSITY Summer 2007
24. Michal Sramka , "DATA MINING AS A TOOL IN PRIVACY-PRESERVING DATA PUBLISHING", Tatra Mt. Math. Publ. 45 (2010), 151–159 DOI: 10.2478/v10127-010-0011-z
25. Tam´asZolt´an GAL and G´abor KOVACS , "Survey on privacy preserving data mining techniques in health care databases" , Acta Univ. Sapientiae, Informatica, 6, 1 (2014) 33–55
26. A N K Zaman and Charlie Obimbo , "Privacy Preserving Data Publishing: A Classification Perspective", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 5, No.9, 2014.
27. Bee-Chung Chen , Daniel Kifer , Kristen LeFevre and AshwinMachanavajjhala
28. Yu Zhu and Lei Liu , "Optimal Randomization for Privacy Preserving Data Mining", KDD'04, August 22–25, 2004, Seattle, Washington, USA. Copyright 2004 ACM 1-58113-888-1/04/0008 ...$5.00.\
29. Yan Z. , Li X. and Zhang P. ,"A Review on Privacy-Preserving Data Mining", December , 2014.