ABSTRACT:

With the growth of the digital world things get easier to share and transfer but this increase the privacy attack, as many data contain different private information of common people. So sharing of data is done by providing the security to the sensitive data that may cause harm or leak the private information of an individual. So the major goal of privacy preserving, is to find the important data then make change to the dataset in order to protect that information from other. In this paper by using Apriori algorithm frequent rules are generate then perturb those rules transaction for reducing the attacker knowledge. This is a reversible process means perturbation will hide information then de-perturbation will regenerate it.

Keywords: Association Rule Mining, Data Perturbation, De-Perturbation Fake Transaction. - Privacy Preserving Data Mining

[1] INTRODUCTION

In this field of data mining different algorithm has develop which are useful in their field but the main purpose of all the work is to retrieve information from the raw data. Here information is find by searching some kind of pattern of data which is already know or information can be generate by generating different patterns in the from the dataset but they are not know but because of their frequency it is found. Mining can be apply in different field like research, marketing analysis, Medical diagnosis, atmosphere forecast etc [1]. So extracting information from the data is done by data mining but this lead to one problem of gathering information from the data in different kind of unsocial activity. This can be understand as if different dataset of the social networking websites are available

Then an intruder can develop information of an individual such as his friend list, his different location, etc. which lead to put him in trouble. So a proper step need to be taken before publishing
any dataset or sending dataset from one place to another, in order to provide privacy of the people from such kind malicious purposes.

Many algorithms and techniques has develop till now for providing the security of the individual in dataset such as perturbation by adding noise, adding fake values, horizontal, vertical partition. So the main purpose of these methods is to develop a kind of confusion or hiding important information from the dataset. By adding fake values overall data may be loss or rules that give information or may help in generating decision also loss. Generally with the help of association rules data mining is done where frequent patterns are develop by using different association rule generating algorithm such as Aprior algo, so in privacy preserving these rules are hide by changing the values of those rules that generate those rules above some threshold value [2].

This can be understand as suppose if we have a dataset that contain individual information its country, salary, occupation, race, etc. then depend on the country name some sensitive rules can be generate that 1. other country people are less salaried , 2. That race of black people are unemployed or given work of low designation only. So hiding of such kind of sensitive information is quit tough by reading individual information or group. So simple thing is to perturb the dataset by different techniques of privacy preserving methods.

[2] Related Work

In [3] An Efficient Algorithm for Privacy Preserving Data Mining Using Heuristic Approach M.Mahendran, Dr.R.Sugumar. Whole work is divide into three steps first is for K-Anonymity because there are many transaction that give direct information of the customer such as the salary of the customer is the information which one need to be hide, then gender and age are also column of the customer. So out of many approaches of hiding this valuable data of the customer one can easily control by making multiple copy of the same data for increasing the confusion and no one get direct information of the customer.

In [4] Enabling Multilevel Trust in Privacy Preserving Data Mining Yaping Li, Minghua Chen, Qiwei Li, and Wei Zhang. They focus on the multilevel party trust by the use of the data perturbation where they create different dataset for the different users of different trust level. They focus on the problem that if multiple users having different copy of the trust level combine there copies then the original dataset can be regenerate[4]. So in order to over come this problem they have given new concept of perturbing the dataset of lower trust from the higher perturbed dataset. In this way if all the lower trust level will combine there dataset then they cannot make the as original dataset as of the just higher level one. Corner-wave Property states that from M perturbed copies, the privacy goal is achieved if the noise covariance matrix $K_{ZZ}$ has the corner-wave pattern as shown in [7]. Specifically, we say that an $M \times M$ square matrix has the corner-wave property if, for every from 1 to $M$, the following entries have the same value as the $(i, j)$ entry. All entries to the right of the $(i, j)$ entry in row $i$. All entries below the $(i,j)$th entry in column $i$.

In [5] Privacy-Preserving Mining of Association Rules From Outsourced Transaction Databases Fosca Giannotti, Laks V. S. Lakshmanan, Anna Monreale, Dino Pedreschi, and Hui (Wendy) Wang Here dataset is use for the privacy is taken from Cooperative customer expenditure. Which has the item index, price, category, etc. In order to put this dataset on the server for different purpose it needs protection from unauthorized user who uses it for unfamiliar
activities. As this dataset need to use by the authorized person as well but the perturbed data is not the correct set for the user to read it, so a successful reading of the authorized user can be possible by a lossless recoverable method. For this method need for perturbing and remove that perturbation from the dataset. Perturbation is done by Cluster of Items: As transaction is a collection of item set that is figure out to make proper co-relation during the perturbation. Frequent Item: Here by A- prior association rule frequent item pattern are find and replace only those by the less frequent set and to remember these transaction one has to generate a sequence that can be obtain by hash function. So special table need to maintain for remembrance at sender, receiver side. Fake transactions: By this fake transaction need to add in the dataset. De-
perturbation: Here as the server get request of the dataset then it pass minimum support value for calculation of original dataset recovery from the perturbed dataset copy. Now this support will specify the item set number to be present in the original dataset and on the basis of this it will remove the reset the transaction as the position is find by the hash function. As many chipper texts is replace original groups of item then replace those with the original one. Here in order to perturb the dataset fake transactions are added which increase the dataset size so the space requirement is too large. There is no focus on the individual information hiding such as age, income, etc.

In [6] k -Support Anonymity Based on Pseudo Taxonomy for Outsourcing of Frequent Item set Mining Chih-Hua Tai, Philip S. Yu, Ming-Syan Chen This paper focuses on outsourcing frequent item set mining and examines the issue on how to protect privacy against the case where the attackers have precise knowledge on the supports of some items. We propose a new approach referred to as k-support anonymity to protect each sensitive item with k−1 other items of similar support. To achieve k-support anonymity, we introduce a pseudo taxonomy tree and have the third party mine the generalized frequent item sets under the corresponding generalized association rules instead of association rules. The pseudo taxonomy is a construct to facilitate hiding of the original items, where each original item can map to either a leaf node or an internal node in the taxonomy tree. The rationale for this approach is that with a taxonomy tree, the k nodes to satisfy the k-support anonymity may be any k nodes in the taxonomy tree with the appropriate supports. So this approach can provide more candidates for k-support anonymity with limited fake items as only the leaf nodes, not the internal nodes, of the taxonomy tree need to appear in the transactions. Otherwise for the association rule mining, the k nodes to satisfy the k-support anonymity has to correspond to the leaf nodes in the taxonomy tree. This is far more restricted. The challenge is thus on how to generate the pseudo taxonomy tree to facilitate k-support anonymity and to ensure the conservation of original frequent item sets.

[3] PROPOSED WORK

It is well known that security of the organization data is very important when it is store at server so there multiple access point can be utilize for updating the data. At the same time it is also required that intruders or attacker may not get that valuable data, but if they get access then it should be secure enough that no fruitful information can be generate by it. Whole work is divide into two part first is perturbation other is de-perturbation. So there are two separate steps for the work, in perturbation original dataset is perturbed in such a way that whole set of transaction is look like original but it is not able to produce any kind of knowledge at all. As the perturbed data
has to perturbed in such a way that it can only hide information but when the user want to read it again then it de-perturbed the data again. So whole work is design in such a way that it get reverse easily.

### A. Perturbation

**Pre-Processing:** Here as the dataset is not available in the processing format, so some step is required to arrange data. This can be understand as the work is focus on the perturbing the text information so the numeric data can be removed from the dataset. Then provide separate column for each item in the dataset.

**B. Association Rule**

As the main perturbation is required for the dataset so dataset shold look like as original dataset but no useful can be generate by mining. So first of all set of association rules need to generate from the dataset which is a combination of different item set. For each rule support value generate this can be understand as: Let A and B are two item if \( A \rightarrow B \) is a rule then support of the rule is the fraction of the total transaction where both A and B present to the total number of transaction in dataset.

\[
\text{Support}(A \rightarrow B) = \frac{A \cup B}{\text{Total Transaction}}
\]

Now by the use of the Aprior algorithm frequent pattern of rules are filter out where a minimum support value is pass so the rules which are above those minimum support value are considered as the hiding rules or hiding pattern. Now for perturbing these rules manipulation need to done where dataset transaction are changed so that. rules which are above minimum support can be changed and only those rules are present in the dataset which are non frequent in item set.

**Multivariate normal cumulative distribution function (Jointly Gaussian Function):** In order to provide different location in the dataset for perturbation one has to generate a series of random values which should be in the range of the total dataset size. Consider \( G_i \) where \( i=1,2,3,4, \ldots \ldots , L \) and L is Gaussian random variables. They are said to be Multivariate normal cumulative distribution function if and only if each of them is a linear combination of multiple independent Gaussian random variables. Equivalently, \( G_1 \) through \( G_L \) are Multivariate normal cumulative distribution function if and only if any linear combination of them is also a Gaussian random variable. A vector formed by Multivariate normal cumulative distribution function variables is called a jointly Gaussian vector. For a Multivariate normal cumulative distribution function

\[
G = |G_1, \ldots , G_L|_T , \text{ its probability density function is as follows: for any real vector } g.
\]

\[
f_G(g) = \frac{1}{\sqrt{\det(K_g)}(2\Pi)^{L/2}} \exp\left(-\left(g - \mu_G\right)^T K_g^{-1} \left(g - \mu_G\right)/2\right)
\]

Where \( \mu_G \) and \( kg \) are the mean vector and covariance matrix of \( G \), respectively. Decision of choosing the transaction number in the database where perturb transaction need to add is random & that is generate by the Gaussian function which take two parameter mean, Co-variance.

Here original dataset is preprocessed then generate the rules by Aprior Algorithm and filter more frequent rules which act as a hidden rule. Then for each hidden rule find number of transaction where those items can be perturbed so that overall support will be reduce. As the change if done...
in sequence forms then dataset starting portion get perturb and rest remain same. So by MNCDF random position is generated and transaction which contain those rules are perturb as well. In this way most of transactions of the dataset are perturbed. Here one important factor of the algorithm is that it will not increase the size of the dataset. Because of this random perturbation which is base on the key it will not break the security of the perturbated transaction.

![Proposed Perturbation Steps](image)

**Figure 1. Proposed Perturbation Steps.**

**C. Proposed Perturbation Algorithm: Here**

1. Original_Dataset $\leftarrow$ Pre_Process(Original_Dataset)
2. $R[n] \leftarrow$ Association_rule(Original_Dataset)  // All frequent Rules
3. Loop 1:n
4. If $R[n] >$ Minimum_Support
5. $FR[m] \leftarrow R[n]$
6. Endif
7. End Loop
8. Position[s] \( \leftarrow \) MNCDF(Key)  \(\times\) Random Position
9. Loop 1:m  \(\times\) Select Rule
10. Fake_item \( \leftarrow \) FR(m)
11. Loop 1:s  \(\times\) Fake Position
12. Pertub_DataSet(s) \( \leftarrow \) Fake_item  \(\times\) Perturb Transaction
13. End Loop
14. End Loop

D. De-perturbation:

When one need to get the dataset back or re-generate then De-perturbation of the perturbed dataset is done. In Which simple pre-processes the dataset as done in perturbation. Here it generate the random position by MNCDF function by passing same KEY value, then de-perturb those transaction position which is specify by the MNCDF function, by replacing the original items.

Proposed De-Perturbation Algorithm

1. Pertub_DataSet \( \leftarrow \) Pre-Process(Pertub_DataSet)
2. Position[s] \( \leftarrow \) MNCDF(Key)  \(\times\) Random Position
3. Loop 1:s
4. Original_Dataset [s] \( \leftarrow \) Pertub_DataSet (s)
5. End Loop

![Diagram](image)

Figure 2. Proposed De- Perturbation Steps.

[4] EXPERIMENT AND RESULT

This section presents the experimental evaluation of the proposed perturbation and de-perturbation technique for privacy prevention. To obtain AR this work used the Apriori algorithm [1], which is a common algorithm to extract frequent rules. All algorithms and utility measures were implemented using the MATLAB tool. The tests were performed on an 2.27 GHz Intel Core i3 machine, equipped with 4 GB of RAM, and running under Windows 7 Professional.
Experiment done on the customer shopping dataset which have collection of items, cost, Total amount, etc. attributes.

A. EVALUATION PARAMETER

Execution time: It is the measure of the algorithm for evaluating the time required for execution. As the server is very important hardware so expectation is of low execution time is more.

DataSet Size: Here size of dataset is analyzed after perturbation. As if the size increases then it require more space to store it on the server.

Originality: As the Perturbation change the original values so, it is necessary to find the original values present in the dataset.

B. RESULT

At the sender side perturbation of the original dataset is done so server will never know the original values of the dataset. In order to compare our proposed work one paper having similar concept are use but it was base on Hash Function and utilize Rob-frugal method for there implementation from [5].

<table>
<thead>
<tr>
<th>Execution Time Perturbation</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Min. Supp</td>
<td>Hash Base</td>
</tr>
<tr>
<td>20</td>
<td>29.8</td>
</tr>
<tr>
<td>16</td>
<td>37.7</td>
</tr>
<tr>
<td>8</td>
<td>40.16</td>
</tr>
</tbody>
</table>

Table 1: Compare Perturbation Execution Time between Hash base [5] and proposed work algorithm

<table>
<thead>
<tr>
<th>Execution Time De- Perturbation</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Min. Supp</td>
<td>Hash Base</td>
</tr>
<tr>
<td>20</td>
<td>2.4</td>
</tr>
<tr>
<td>16</td>
<td>2.62</td>
</tr>
<tr>
<td>8</td>
<td>2.8</td>
</tr>
</tbody>
</table>

Table 2: Compare De-Perturbation Execution Time between Hash base [5] and proposed work algorithm

<table>
<thead>
<tr>
<th>DataSet Size</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Min. Supp</td>
<td>Hash Base</td>
</tr>
<tr>
<td>20</td>
<td>12000</td>
</tr>
<tr>
<td>16</td>
<td>15603</td>
</tr>
<tr>
<td>8</td>
<td>16690</td>
</tr>
</tbody>
</table>
Table 3: Compare Dataset Size between Hash base [5] and proposed work algorithm

![Evaluation Results](image)

Figure 3. Compare Perturbation Execution Time between Hash base [5] and proposed work algorithm.

As above results shown in [Table-1], [Table-2], [Table-3] and [Figure-3] that with the increase in support value the perturbation rules will be decrease so the affecting transaction will also be reduce. This will reduce the overall work and time require for perturbation is lesser. Same in the case of De-perturbation as with the increase in the support value the less perturbed transaction will get original quickly and reduce the de-perturbation time. Comparison between Hash base method [5] and proposed work shows that Both the perturbation and De-perturbation time is less as compare to [5] method.

Now in table three with the decrease in support value Hash base method increase the dataset size, while the proposed work will remain constant. This is because as the dataset size in case of hash method increase because of fake transaction are add in the original dataset while order is present in set. But in case of proposed work fake items replace the original items so transaction number remain same.

[5] CONCLUSION

With the increase in the technology it is required to provide more security for the data present on the server. With this aim paper work for providing different security measures in the proposed work, by utilizing Aprior, MNCDF, methods. Results shows that algorithm is efficient in terms of both space and time complexity. There is always work remain for each field here, one can improve work by reducing perturbing time.

REFERENCES


