Journal of Basic and Applied Engineering Research
Print ISSN: 2350-0077; Online ISSN: 2350-0255; Volume 1, Number 6; October, 2014 pp. 27-32
© Krishi Sanskriti Publications
http://www.krishisanskriti.org/jbaer.html

Privacy Preserving Utility Mining Using Sanitization Approach

Deepika Shrivastava[1], **Rahul Shukla**[2]

[1,2]*Department of CSE, College of Science & Engineering, Jhansi, India*

Abstract: This thesis is basically designed for privacy preserving utility mining using sanitization approach. In this work itemsets are provided safety using an approach , firstly we will calculate the utility of all itemsets as the product of item cost and its number of transactions, then we will set a threshold utility which will be the average of max and min utility. Now, we will try to reduce the difference between the utility of item and threshold, this can be done by applying a formula generated in new algorithm named as "Privacy Preserving utility Mining Using Sanitization(PPUMUS)" developed by us. Applying this algorithm the difference gets reduce to such an extent now those sensitive items which had greater utility than threshold, cannot be mined. Further in this approach, apply some sort of encryption on the item name so that they appear unintelligent to other users and outsiders and provide password protection. The output of both the approaches is a sanitized database DB'.

Keywords: Utility based data mining, sanitization approach, SHA algorithm, Fast utility frequent mining algorithm.

1. INTRODUCTION

To find the right balance between maximizing analysis results that are useful for the common good and keeping the inferences that disclose private information is the aim of privacy preserving data mining. In many cases, sensitive data can be inferred from non sensitive data based on some knowledge and/or skilful analysis. The problem of protection against inference has been addressed in the literature of statistical databases since last decade. However, in the field of data mining and in particular for the task of association rules the focus has been more specific. Here, some researchers refer to the process of protection against inference as data disinfect. Data sanitization is defined as the process of making sensitive information in non production databases safe for wider visibility. Others advocate a solution based on collaborators mining independently their own data and then sharing some of the resulting patterns. This second alternative is called rule disinfect. In this later case, a set of association rules is processed to block inference of so called sensitive rules about organizations or individuals at a minimum.

Privacy: Privacy is defined in the cryptographic community limits the information that is leaked by the distributed computation to be the information that can be learned from the designated output of the computation. Although there are several variants of the definition of privacy, for the purpose of this discussion we use the definition that compares the result of the actual computation to that of an "ideal" computation:

Consider first a party that is involved in the actual computation of a function (e.g. a data mining algorithm). Consider also an "ideal scenario", where in addition to the original parties there is also a "trusted party" who does not deviate from the behavior that we prescribe for him, and does not attempt to cheat. In the ideal scenario all parties send their inputs to the trusted party, who then computes the function and sends the appropriate results to the other parties. Loosely speaking, a protocol is secure if anything that an adversary can learn in the actual world it can also learn in the ideal world, namely from its own input and from the output it receives from the trusted party. In essence, this means that the protocol that is run in order to compute the function does not leak any "unnecessary" information. As an example for the definition of privacy, consider the following problem. Rahul and Ram are managers both working in the same firm, and each of them suspects that one specific servant has stolen something. None of them is completely sure, though, about the identity of the thief, and they would therefore like to compare the names of their two suspects. Since they care about their servants privacy they wish that, (1) if they both have the same suspect, then they should learn his or her name, but (2) if they have different suspects then they should learn nothing beyond that fact. They therefore have inputs x and y, and wish to compute f(x, y) which is defined as 1 if x = y and 0 otherwise. (Note that if f(x, y) = 0 then each party does learn some information, namely that the other party's suspect is different than his/hers, but this is inevitable).

Data privacy in data mining: To establish a good decision tree, we need a pool of training samples. For most cases, real data are collected from individuals for statistical utilities. Even if explicit identification information, e.g. names, can be removed for classification data mining, identities are traceable by matching individuals with a combination of non-identifying information such as date and place of birth, gender, and employer. In addition to storing the samples securely, the private information (particularly that which is medical or financial in nature) of those information providers must be kept in a sanitized version to prevent any kind of privacy leakage. The imperatives of data utility and confidentiality make privacy preservation an important field of research. In Privacy Preserving Data Mining: Models and Algorithms, Agarwal and Yu[1][2] classify privacy preserving data mining techniques, including data modification, cryptographic, statistical, query auditing and perturbation-based strategies. Cryptographic, statistical and query auditing techniques are related to multi-party data mining protocol, inference control and security assurance, all of which are subjects outside of the focus of this thesis. In this chapter, we explore the privacy preservation techniques used by data modification and perturbation-based approaches, and summarize them in relation to decision-tree data mining. The principal attention to Privacy Preserving Data Mining (PPDM) is development of those algorithms, which - by protecting existed private data and knowledge in datasets and accessing the valid results of data mining-provide the possibility to share the critical and private data for analytical aims.

There are two general scenarios in Privacy Preserving Data Mining: the Multi-party collaborations scenario and Data publishing scenario. In the former, the collection of data is distributed between two or more sites, each one owns a part of the private data and these sites collaborate to compute a data mining algorithm on the union of their databases without revealing the data at their individual sites and the results of data mining will only be revealed. The major approach for this scenario is the Secure Multi-party Computation.

In Data publishing scenario the owners or data providers are publishing or sharing their data to acquire data mining results and /or joining the data mining process. the privacy preservation techniques are applied during the data integration or before sending data to the data miner Principal approaches in this scenario based on the goal of privacy preservation-classified in two categories:

FUFM (Fast Utility-Frequent Mining) algorithm: In this paper we conclude that Utility-based data mining is a new research area interested in all types of utility factors in data mining processes and targeted at incorporating utility considerations in both predictive and descriptive data mining tasks. High utility item set mining is a research area of utility based descriptive data mining, aimed at finding item sets that contribute most to the total utility. A specialized form of high utility item set mining is utility-frequent item set mining, which – in addition to subjectively defined utility – also takes into account item set frequencies. This paper We study a novel efficient algorithm FUFM (Fast Utility-Frequent Mining) which finds all utility-frequent item sets within the given utility and support constraints threshold. It is faster and simpler than the original 2P-UF algorithm (2 Phase Utility-Frequent), as it is based on efficient methods for frequent items et mining. Experimental evaluation on artificial datasets show that, in contrast with 2P-UF, our algorithm can also be applied to mine large databases.

```
Algorithm: FUFM
Input:
- database DB
- constraints minUtil and minSup
Output:
- all utility-frequent item sets
[1] L = 1
[2] find the set of candidates of length L with support >= minSup
[3] compute exteded support for all candidates and output utilityfrequent item sets
[4] L += 1
[5] use the frequent item set mining algorithm to obtain new set of frequent candidates of length L from the old set of frequent candidates
[6] stop if the new set is empty otherwise go to [3].
```

2. RELATED WORK

Privacy Preserving Data Mining is a relatively new research area that aims to prevent the violation of privacy that might result from data mining operations on data sets. PPDM algorithms modify original data sets so that privacy is preserved even after the mining process is activated, while minimally affecting the mining results quality. Verykios et al.[4] classified existing PPDM approaches based on five dimensions:

1. Data Distribution, referring to whether the data are centralized or distributed;
2. Data Modification, referring to the modifications performed on the data values to ensure privacy.
   There are different possible operations such as aggregation (also called generalization) or swapping;
3. Data Mining algorithms referring to the target DM algorithm for which the PPDM method is defined.
4. Data or rule hiding referring to whether the PPDM method hides the raw or the aggregated data[7]; and finally,
5. Privacy preservation, referring to the type of technique that is used for privacy preservation: heuristic, cryptography; or reconstruction- based (i.e., perturbing the data and reconstructing the distributions to perform mining).

1. **Utility Mining:** The traditional ARM (Association Rule Mining) approaches consider the utility of the items by its presence in the transaction set. The frequency of item set is not sufficient to reflect the actual utility of an item

set. For example, the sales manager may not be interested in frequent item sets that do not generate significant profit. Recently, one of the most challenging data mining tasks is the mining of high utility item sets efficiently[8]. Identification of the item sets with high utilities is called as Utility Mining. The utility can be measured in terms of cost, profit or other expressions of user preferences. For example, a computer system may be more profitable than a telephone in terms of profit.

Utility mining model was proposed in to define the utility of item set. The utility is a measure of how useful or profitable an item set X is. The utility of an item set X, i.e., u(X), is the sum of the utilities of item set X in all the transactions containing X. An item set X is called a high utility item set if and only if u(X) >= min_utility, where min_utility is a user defined minimum utility threshold [8].The main objective of high-utility item set mining is to find all those item sets having utility greater or equal to user-defined minimum utility threshold. Each row is a transaction. The column represents the no of items in a particular transaction .TID is the transaction identification number

Table 1.1-Transaction Table

TID	A	B	C	D	E
T1	0	0	18	0	1
T2	0	6	0	1	1
T3	2	0	1	0	1
T4	1	0	0	1	1
T5	0	0	4	0	2
T6	1	1	0	0	0
T7	0	10	0	1	1
T8	3	0	25	3	1
T9	1	1	0	0	0
T10	0	6	2	0	2

The sanitizing algorithms in major can be divided into two classes:

Data sharing approach and pattern sharing approach, as can be showed in figure 3. In the former, the sanitization process acts on the data to remove or hide the group of restrictive association rules that contain sensitive knowledge. In the latter the sanitizing algorithm acts on the rules mined from a database instead of the data itself.

Table 1.2- External Utility (Profit) of various items

ITEM	PROFIT($)(PER UNIT)
A	3
B	10
C	1
D	6
E	5

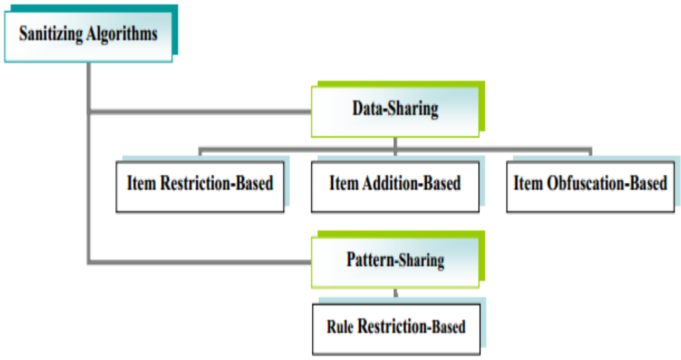

Fig 1.2 A taxonomy of sanitization Algorithm

Among the algorithms of Data – sharing approach, they are classified the following categories : Item restriction-Based, and Item obfuscation- Based.

Item Restriction- Based: These algorithms remove one or more items from a group of transactions containing restrictive rules. In doing so, the algorithms hide restrictive rules by reducing either their supports or confidences below a privacy threshold.

Item Addition – Based: Unlike the algorithms[3], item addition algorithms modify existing information in transaction databases by adding some items not originally present in some transaction. This approach[6] may generate artificial association rules that would not exist in the original database.

Item obfuscation – Based : The algorithms hide rules by placing a mark’’?’’ (unknowns) in items of some transactions containing restrictive rules, instead of deleting such items. In doing so, these algorithms obscure a given set of restrictive rules by replacing known values with unknowns. This approach can apply to medical applications to replace a real value by an unknown value instead of placing a false value. For example, GIH Algorithm proposed by saygin et al.[4] .Regarding pattern- sharing techniques, the only known approach that falls into this category was introduced .

Rule Restriction- Based: This approach blocks some inference channels to ensure that an adversary cannot reconstruct restrictive rules from the non – restrictive ones. In doing so, we can reduce the inference channels and minimize the side effect. For example, DSA Algorithm proposed by oliveira et al.[3]. According to the downward closure property of apriori, we have identified some attacks against sanitized rules, as follows[3]:

Forward – Inference Attack: Let us consider the frequent item set graph in Figure 4. Suppose we want to sanitize the restrictive rules derived from the item set ACD. The naïve approach only removes the item set ACD. Is frequent item set from the released database? In order to deal with this attack, we have to remove at least one subset of ACD in the level 1 of the frequent item set graph during the sanitization process.

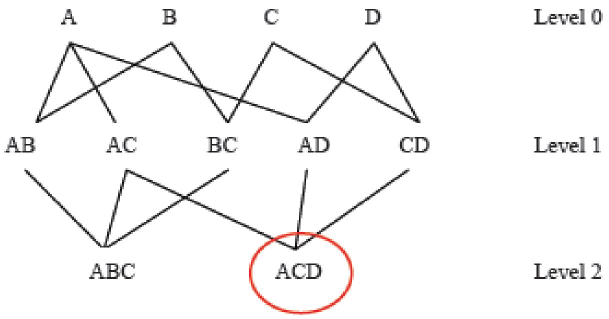

Fig 1.3 An example of forward interference

Backward-Inference Attack: According to Figure 5, suppose we want of sanitize any rule derived from the item set AC. However, if we only remove AC, it is straightforward to deduce the rules mined from AC Since either ABC or ACD is frequent item set. In order to block this attack, we have to remove any superset that contains AC in the transformation process. In this especially case, ABC and ACD must be removed at the same time.

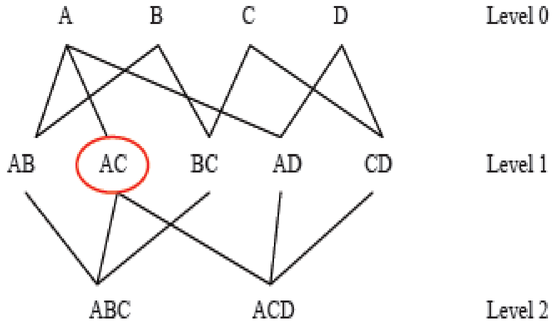

Fig. 1.4-An example of Backward-Inference

2. **The Sanitization Methodology:** Oliveria et al. first proposed the concept of the privacy threshold φ in [5]. The proportion of restrictive patterns that are still discovered from the sanitized database can be controlled by users with privacy threshold φ, and this proportion ranges from O% to 100% When φ = 0%, no restrictive patterns are allowed to be discovered. When φ = 0%,there are no restrictions on the patterns. In order works, all restrictive patterns. In other words, all restrictive patterns can be discovered. The advantage of having this threshold is that between privacy and the disclosure of information can be balanced.

2.1 The General Approach

Our heuristic approach for the privacy preserving utility mining consists of two processes: (1) identifying the sensitive item sets (2) modifying transaction containing the sensitive item sets.

Step 1: Identifying the sensitive item sets. By applying any utility mining algorithm, for example: Two-Phase Algorithm [9], the database owner first discovers all high utility item sets from the original database with a specified utility threshold. The owner must clearly know what knowledge he wants to protect. That is, the owner must specify which sensitive item sets should be protected and will not be mined by the data receiver from the released database.

Step 2: Modifying transaction containing sensitive item sets. For each sensitive item sets, we modify the quantity of items in some transactions containing the sensitive item set, until the utility of the sensitive item set is less than the given minimum utility threshold.

3. MOTIVATION AND APPLICATION

Data mining, the extraction of hidden predictive information from large databases, is a powerful new technology with great potential to help companies focus on the most important information in their data warehouses. Data mining tools predict future trends and behaviors, allowing businesses to make proactive, knowledge-driven decisions. The automated, prospective analyses offered by data mining move beyond the analyses of past events provided by retrospective tools typical of decision support systems. Data mining tools can answer business questions that traditionally were too time consuming to resolve. They scour databases for hidden patterns, finding predictive information that experts may miss because it lies outside their expectations. Most companies already collect and refine massive quantities of data. Data mining techniques can be implemented rapidly on existing software and hardware platforms to enhance the value of existing information resources. The growing scope of company and corporate enviroment is having cut throat competition as a result information leakage, pattern recognition by opponents , all such adversaries gave rise to the significance of privacy preserving data mining.

4. PROPOSED ALGORITHM

As per the proposed methodology privacy preserving data mining in done in three major steps:

Step 1: The first step is to differentiate between the database administrator and other employees, that is a separate password for everyone involved directly with the company. So a program segment is involved which generates password, which gets expired every month and then new passwords are allotted. This is under the authority of DBA.

Step 2: For the original database DB, available to the DBA , a sanitized database DB' is generated for all other employees. To do so:

- Input: the original database DB; the minimum utility threshold ε; the sensitive itemsets U={S1, S2, . . . , Sn}.
- Output: the sanitized database DB' so that Si cannot be mined.
- Step 2.1: For each sensitive itemset Si ε U
- Step 2.2: diff = u(Si) - ε // the utility value needs to be reduced
- Step 2.3: while (diff > 0) {
- Step 2.4: vv = (ε*100)/u(Si); //(threshold*100)/ utility of item.
- Step 2.5: modify o(ip,t q) with o(ip,t q) = (diff*vv)/100;
- Return sanitized database DB'.

Step 3: To the result of above step add some more steps:

- Step 3.1: Encrypt the domain of product name.
- Step 3.2: Delete the column of number of transactions(sales).

In this algorithm vv stands for virtual value by which original utility is being modified and o(ip,tq) is the modified utility of that sensitive item. As we know, Data sanitization is the process of altering the transactions. To do so, a small number of transactions have to be modified by deleting one or more items from them or even changing items in transactions, i.e., adding noise to the data. Now, this resulting sanitized database is only allowed to be viewed to other employees not the original one, so that sensitive items(having utility > threshold) cannot be mined. As we have encrypted the product name and deleted the sales quantity , this won't allow employee himself to generate patterns which are beneficial to the company. Along with this process, the special password protection which changes frequently, and doesn't allow any cheat.

5. EXPERIMENTAL ANALYSIS

To measure the effectiveness, we adopt the set of metrics proposed in terms of information loss and non-sensitive patterns removed as a side effect of the transformation process. The performance measures are specified as follows:

(a) Hiding failure (HF): the ratio of sensitive item sets that are disclosed before and after the sanitizing process. The hiding failure is calculated as follows:

$$\mathrm{HF} = \frac{|U(DB')|}{|U(DB)|}$$

where U(DB)and U(DB') denote the sensitive item sets discovered from the original database DB and the sanitized database DB' respectively.

(b) Miss cost (MC): the difference ratio of legitimate item sets found in the original and the sanitized databases. The miss cost is measured as follows:

$$\mathrm{MC} = \frac{|\sim U(DB) - \sim U(DB')|}{|\sim U(DB)|}$$

Table 1.3: Calculated Value of HF and MC.

Threshold utility ε	HHUIF & MSICF hiding failure	HHUIF missing cost	MSICF missing cost	Resulting hiding failure	Resulting missing cost
2000	0.00	0.00	0.00	0.00	0.66
3000	0.00	0.93	12.96	0.00	0.56
4000	0.00	62.04	70.37	0.00	0.41

6. CONCLUSION & FUTURE WORKS

This algorithm is based on modifying the database containing the sensitive itemsets so that the utility value can be reduced below Utility threshold value. There is no possible way to reconstruct the original database from the Sanitized one. In our experimental results, PPDMUS has the lower miss costs in randomized datasets. The two sanitization approaches namely HHUIF and MSICF for privacy preservation gave meaningful results but my approach or algorithm is comparatively better in the sense of all metrics i.e., lowest hiding failure and lowest missing cost. In addition to sanitized DB , this work involves security feature which makes it more efficient. The result of this work is a complete package of fully secured, privacy preserved information system. In the future, a more superior sanitization algorithms can be developed to minimize the impact on the sanitized database in the process of hiding sensitive itemsets. The work can also be expanded with a probabilistic to supplement the empirical, which require further exploration.

REFERENCES

[1] R.Agrawal, T. Imielinski, and A. N. Swami, "Mining association rules between sets of items in large databases," In Proceedings of the 1993 ACM SIGMOD International Confe4rence on Management of Data, pp. 207-216, Washington, 1993.

[2] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules." In Proceedings of 20th International conference on Very Large Data Base, pp. 487-499, Morgan Kaufmann, 1994.

[3] S.R.M. Oliveira, O.R. Zaiane, and y. Saygin, "Secure association rule sharing," In Proceedings of 8th Pacific-Asia Conference on Knowledge Discovery and Data Mioning (PAKDD'04), pp. 74-85, Australia, May 2004.

[4] Y. Saygin, V.S. Verykios, and C. Clifton, "Using unknowns to prevent discovery of association rules," SIGMOD Record, vol. 30, no. 4, pp. 45-54, 2001.

[5] B. Barber, and H.J. Hamilton, "Extracting share frequent item sets with infrequent subsets," Data Mining and Knowledge Discovery, vol. 7, no.2, pp. 153-185, April 2003.

[6] S. Rizvi, and J. Haritsa, "Maintaining data privacy in association rule mining," In Proceedings of 28th Intl. Conf. on Very Large Databases (VLDB), August 2002.

[7] S.J. Rizvi and J.R. Haritsa, "Privacy-preserving association rule mining," In Proceedings of the 28th Int'l Conference on Very Large Databases, August 2002.

[8] H. Yao, H.J. Hamilton, and C.J. Butz, "A foundational approach to mining item set utilities from databases," In Proceedings of the 4th SIAM international Conference on Data Mining, Florida, USA, 2004.

[9] A Two-Phase Algorithm for Fast Discovery of High Utility Item sets by Ying Liu, Wei-keng Liao, and Alok Choudhary Electrical and Computer Engineering Department, Northwestern University, Evanston, IL, USA 60208”