**IOSR Journal of Engineering (IOSRJEN)** **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
Vol. 04, Issue 07 (July. 2014), ||V1|| PP 36-41

# Improved Association Rule Hiding Algorithm for Privacy Preserving Data Mining

Hiren R. kamani, Supriya Byreddy
*M.Tech Scholar, Computer Engineering School of Engineering, R K University Gujarat, India*
*Assistant Professor, Computer Engineering School of Engineering, R. K. University Gujarat, India*

***Abstract*: -** the main objective of data mining is to extract previously unknown patterns from large collection of data. With the rapid growth in hardware, software and networking technology there is outstanding growth in the amount data collection. Organizations collect huge volumes of data from heterogeneous databases which also contain sensitive and private information about and individual .The data mining extracts novel patterns from such data which can be used in various domains for decision making .The problem with data mining output is that it also reveals some information, which are considered to be private and personal. Easy access to such personal data poses a threat to individual privacy. There has been growing concern about the chance of misusing personal information behind the scene without the knowledge of actual data owner. Privacy is becoming an increasingly important issue in many data mining applications in distributed environment. Privacy preserving data mining technique gives new direction to solve this problem. PPDM gives valid data mining results without learning the underlying data values .The benefits of data mining can be enjoyed, without compromising the privacy of concerned individuals. The original data is modified or a process is used in such a way that private data and private knowledge remain private even after the mining process. The objective of this paper is to implement an improved association rule hiding algorithm for privacy preserving data mining. This paper compares the performance of proposed algorithm with the two existing algorithms namely ISL, DSR and WSDA.

***Index Terms: -*** *Privacy Preservation Rule Mining, Sensitive Data, association rule hiding.*

## I. INTRODUCTION

The data mining technologies have been an important technology for discovering previously unknown and potentially useful information from large data sets or databases. They can be applied to various domains, such as Web commerce, crime reconnoitering, health care, and customer's consumption analysis. However, the technologies can be threats to data privacy. Association rule analysis is a
Powerful and popular tool for discovering relationships hidden in large data sets. Some private information could be easily discovered by this kind of tools. Therefore, the protection of the confidentiality of sensitive information in a database becomes a critical issue to be resolved.

Here before collaborating/releasing the dataset to the other party, each party is willing to hide sensitive association rules of its own sensitive products/data. So, the sensitive information (or knowledge) will be protected. In 1999 first time Atallah *et al.* Proposed association rule hiding problem in the area of privacy preserving data mining [2].

Privacy preserving data mining (PPDM) is considered to maintain the privacy of data and knowledge extracted from data mining. It allows the extraction of relevant knowledge and information from large amount of data, while protecting sensitive data or information. To preserve data privacy in terms of knowledge, one can modify the original database in such a way that the sensitive knowledge is excluded from the mining result and non sensitive knowledge will be extracted. In order to protect the sensitive association rules (derived by association rule mining techniques), privacy preserving data mining include the area called “association rule hiding”. The main aim of association rule hiding algorithms is to reduce the modification on original database in order to hide sensitive knowledge, deriving non sensitive knowledge and do not producing some other knowledge.

In this paper, we propose an improved algorithm, for hiding sensitive association rules. The algorithm can completely hide any given sensitive rule. Experimental results show that this algorithm performs well then the previous works done in ISL, DSR and WSDA, in terms of execution time and side effects generated.

Rest of this paper is organized as follows: - In Section 2, discusses Previous work carried out in this field. The Problem formulations and notations are given in section3. Section 4 presents our association rule hiding approaches by identifying open challenges. Designing of algorithm given in section 5. Results and simulations given in Section 6, Section 7 concludes my study by identifying future work with references at the end.

## II. PREVIOUS WORK

There are various methods available for association rule hiding Verykios et al. [3] suggest Data-Distortion technique a sub class of Heuristic based approaches. It changes a selected set of 1-values to 0- values (delete items) or 0-values to 1- values (add items), Y. Saygin et al.[4][5] were the first to propose blocking technique in order to increase or decrease the support of the items by replacing 0's or 1's by unknowns "?". This is again a sub class of Heuristic based approaches. Vaidya and Clifton [6] proposed a secure approach using Cryptography based approach for sharing association rules when data are vertically partitioned. The authors in [7] addressed the secure mining of association rules using Cryptography based approach over horizontal partitioned data. Border based approach uses the theory of borders presented in [8]. These approaches pre-process the sensitive rules so that minimum numbers of rules are given as input to hiding process. The sensitive association rules are hidden by modifying the borders in the lattice of the frequent and the infrequent item set of the original database. The item sets which are at the position of the borderline separating the frequent and infrequent item sets forms the borders. So, they maintain database quality while minimizing side effects. Gkoulalas and Verykios [9] proposed an approach to find optimal solution for rule hiding problem which tries to minimize the distance between the original database and its sanitized version. The authors in [10] proposed a novel, exact border-based approach that provides an optimal solution for the hiding of sensitive frequent item sets by minimally extending the original database by a synthetically generated database part - the database extension.

R.Natarajan, Dr.R.Sugumar, M.Mahendran, K.Anbazhagan[1] suggest a new association rule hiding algorithm for hiding sensitive items in association rules, In this proposed algorithm, a rule X → Y is hidden by decreasing the support value of X U Y and increasing the support value of X. That can increase and decrease the support of the LHS and RHS item of the rule correspondingly.

## III. PROBLEM FORMULATION AND NOTATIONS

In Table 1, we summarize the notations used hereafter in this paper. The support of item set S can be computed by the following equation:

$$\text{Support}(S) = \|S\| / |D|, \qquad (1)$$

Where $\|S\|$ denotes the number of transactions in the database that contains the item set S, and $|D|$ denotes the number of the transactions in the database D. We call S as a frequent item set if support(S) ≥ min_support, a given threshold. A transaction ti supports S, if $S \subseteq ti$. An association rule is an implication of the form $X \rightarrow Y$, where $X \subset I$, $Y \subset I$ and $X \cap Y = \emptyset$. A rule $X \rightarrow Y$ is strong if

1) Support $(X \rightarrow Y) \geq$ min_support and
2) Confidence $(X \rightarrow Y) \geq$ min_confidence,

where min_support and min_confidence are two given minimum thresholds, and the support$(X \rightarrow Y)$ and confidence$(X \rightarrow Y)$ can be computed by the following equations:

$$\text{Support } (X \rightarrow Y) = \|X \cup Y\| / |D|; \qquad (2)$$

$$\text{Confidence } (X \rightarrow Y) = \|X \cup Y\| / | X |. \qquad (3)$$

Table 1. Notations and Definitions

| | |
|---|---|
| I | I = {i1, i2, ..., im} a set of items in a transaction database |
| D | The original database D = {t1, t2… tn}, where every transaction ti is a subset of I, i.e., ti⊆I. |
| D' | the sanitized database which is transformed from D |
| X | Set of Sensitive Rule |
| T | transaction belongs to D |
| Ti. k | k item from ti transaction |

Example 1. An example database is shown in Table 2. There are nine items, |I|=9, and five transactions, |D|=5, in the database. Table 3 shows the frequent item sets generated from Table 2 for min_support = 60%. For the example S = {1, 4, 7}, since S⊆t1, S⊆t2 and S⊆t3, we obtain ||S||=3. Therefore, support (1, 4, 7) = ||S|| / |D| = 60%. Table 4 shows the association rules generated from Table 2 for min_support = 60% and min_confidence = 75%. For the example rule 1,4→7, since ||{1,4}|| = 3 and ||{1,4,7}||=3, with the equations (2) and (3), we can get support(1,4→7) = 60% and
Confidence (1, 4→7) = 100%.

Our study goal is to completely hide all sensitive rule while minimizing the side effects generated from the database modification.

Table 2. Set of transactional data

| TID | ITEMS |
|---|---|
| 1 | 1,2,4,5,7 |
| 2 | 1,4,5,7 |
| 3 | 1,4,6,7,8 |
| 4 | 1,2,5,9 |
| 5 | 6,7,8 |

Table 3. Association rules generated from Table 2, min_support=60% and min_confidence=75%

| | |
|---|---|
| 1→ 4 (60%, 75%) | 4, 7→ 1 (60%, 100%) |
| 7→ 4 (60%, 75%) | 1→ 7 (60%, 75%) |
| 4→ 1 (60%, 100%) | 1→ 4, 7 (60%, 75%) |
| 1, 4→ 7 (60%, 100%) | 7→ 1 (60%, 100%) |
| 1→ 5 (60%, 75%) | 4→ 1, 7 (60%, 100%) |
| 1, 7→ 4 (60%, 100%) | 4→ 7 (60%, 100%) |
| 5→ 1 (60%, 100%) | 7→ 1, 4 (60%, 75%) |

## IV. OUR APPROACH

In this approach we focused upon specific transaction such that it's one of the item has highest weight, here weight can be defined as maximum number of rule R belongs to X, supported by transaction item ti.k. As well as that transaction has less no of item. Using this process we are able to short list a set transactions, which are more likely to do modification.

## V. DESIGNING OF ASSOCIATION RULE HIDING ALGORITHM

We now demonstrate the proposed algorithm given D original database, X set of sensitive rules, minimum_support, minimum_confidance. Goal of this algorithm is to generate sanitized database D', where all sensitive rule hidden. In table 4 proposed algorithm pseudo code is available.

As suggested in pseudo code first of all calculate the maximum weight associated with each transaction, here weight can be calculated by maximum number of sensitive rule support by transaction item / pow(Ti.length-1).

Let's assume we want to hide rule 1 5→ 7, then first of all identify the weight of each transaction for example for transaction t1={1,2,4,5,7}, weight associated with item 1, 5 and 7 is 1 so maximum of it is 1 and transaction length t1 is 5 so finally we have weight associated transaction t1 is 1/16. The weight associated with each transaction is mentioned in table 6.

Here from table 6 we can identify most likely transaction for modification to hide the sensitive rule. By arranging this transaction in descending order we have transaction no 2 for modification because transaction 5 does not contain the any item Ti.k that belongs to X, so next transaction t2 is chosen, here Ti.K=1 is belongs to X, so it is removed from transaction now after calculating minimum_support of rule 1 5→ 7, it is less than the given minimum_support so now we have nothing left in X. hence newly generated database D' does not contain sensitive rule.

Table 4. Association Rule Hiding Algorithm

```
Input : A source database D, Minimum_Support,
Minimum_Confidence, X set of Hidden Rules
Output: D' sanitized database where all rule belongs to X
is completely hidden.

1.    Being
2.    Compute weight for each transaction
2.1.  For Every transaction Ti belongs to D
2.2.  F or each Sensitive Rule Xj belongs to X Do
2.3.  If Xj Supported by Ti then
2.4.  Weight = maximum no. of rule supported by Ti.k in X
      / pow(2,Ti.length-1)
2.5.  Store weight along with associated transaction
3.    While X is not Empty Do
3.1.  Select transaction Tm having maximum weight.
3.2.  Select item from transaction Tm which is having
      highest weight Tm.k
3.3.  If Support Xj >= minimum_Support and Tm.k belongs
      to Xj Then
 Remove Tm.K
Else
Skip the Tm
4.    If support(Xj)< minimum_Support or
      Confidence(Xj)<minimum_Confidence  Then
Remove Xj from X
End while;
5.    End
```

Table 6. Weight Associated with transaction

| TID | Weight |
|---|---|
| 1 | 1/16 |
| 2 | 1/8 |
| 3 | 1/16 |
| 4 | 1/8 |
| 5 | 1/4 |

## VI. SIMULATIONS AND RESULTS

We have used weka for analysis purpose as well as our couple of code.
Here we have showed three comparison charts as follows,
1) Time required for Hiding process
2) No of Entry Modified during Hiding
3) No of Lost Rule after hiding process.

Evaluation Matrix 1: Time Complexity
The first experiment shows the relationship between CPU time and number of transactions. Table 7 shows the experimental results. In this experiment, the Minimum confidence value is set 60% and minimum support values are taken as 40% for 1000, 2000 and 3000 transactions respectively.

Table 7. CPU Time Utilization

| Number of Transaction | CPU Time(milliseconds) | | | |
|---|---|---|---|---|
| | ISL | DSR | WSDA | Proposed Algorithm |
| 1000 | 842 | 688 | 425 | 83 |
| 2000 | 1655 | 1337 | 827 | 120 |
| 3000 | 2567 | 2153 | 1273 | 150 |

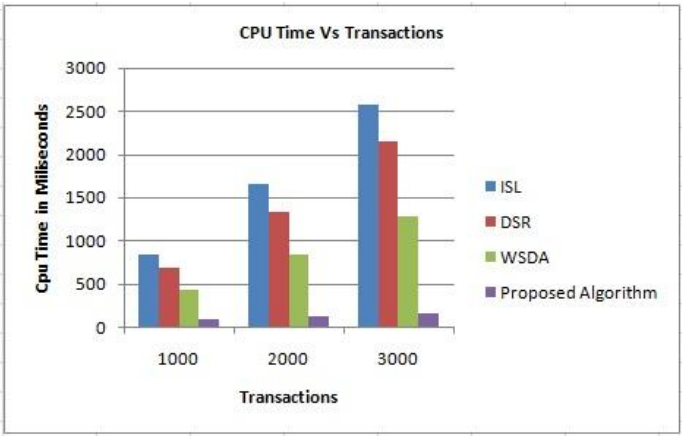


Figure 1. CPU Time Vs Transactions

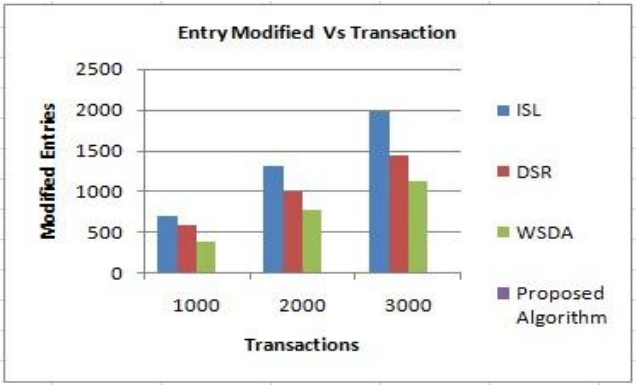
Evaluation Matrix 2: Number of Entry Modified

This experiment shows the relationship between No of Entry modified and number of transactions Table 8 shows the experimental results. In this experiment, the Minimum confidence value is set 60% and minimum support values are taken as 40% for 1000, 2000 and 3000 transactions respectively.

Table 7. Number of modified entries

| Transaction | No. Entry Modified | | | |
|---|---|---|---|---|
| | ISL | DSR | WSDA | Proposed Algorithm |
| 1000 | 683 | 575 | 372 | 3 |
| 2000 | 1297 | 982 | 764 | 2 |
| 3000 | 1980 | 1442 | 1127 | 10 |

Figure 2. Entry Modified Vs Transactions



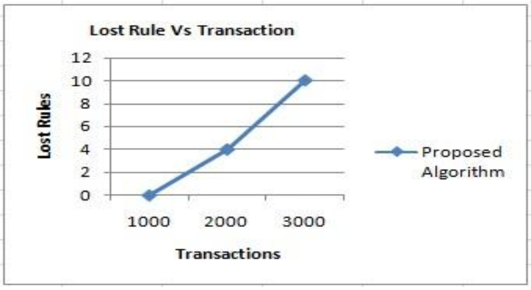
Evaluation Matrix 3: Number of New Rule Generated

This experiment shows the relationship between No of New rule generated and number of transactions Table 8 shows the experimental results. In this experiment, the Minimum confidence value is set 60% and minimum support values are taken as 40% for 1000, 2000 and 3000 transactions respectively.

Table 8. Number of Lost Rule

| Transaction | No. Lost Rule |
|---|---|
| 1000 | 0 |
| 2000 | 4 |
| 3000 | 10 |

Figure 3. Lost Rule Vs Transactions



After review of experimental result we have been observed first characteristic is less modification in database. Table 7 shows the relationship between total number entries modified and number of transaction; the proposed algorithm modified a few numbers of entries for hiding a given set of rules in all the datasets.

The second characteristic has observed is the CPU time requirement. Table 6 shows the relationship between Total CPU time for number of entries modified and number of transaction, the proposed algorithm modified a few CPU time for hiding rule and modified entries are given set of rules in all the datasets.

And last characteristic that we observer is regarding number of lost rule after hiding process table 8 shows the relationship between Number of rule and number of lost rule.

## VII. CONCLUSION

Privacy preserving data mining is a new body of research focusing on the implications originating from the application of data mining algorithms to large public databases. In this study, we have delved into the deep waters of knowledge hiding, which is primarily concerned with the privacy of knowledge that is hidden in large databases. More specially, we have surveyed a research direction that investigates how sensitive association rules can escape the scrutiny of malevolent data miners by modifying certain values in the database. We have also presented a thorough analysis and comparison of the surveyed approaches, as well as a classification of association rule hiding algorithms to facilitate the organization in our presentation. Before we conclude our study we have provided a comparisons of other related hiding approaches like ISL, DSR, WSDA and we have introduced a set of metrics for the evaluation of the association rule hiding algorithms. Moreover, we strongly believe that the emergence in the association rule hiding area will come into play in the evolution of other related fields in data mining and will cause new waves of research study. At that point, we will be certain that our expectations regarding the destiny of this field will have been fulfilled.

## REFERENCES

[1] R.Natarajan, Dr.R.Sugumar, M.Mahendran, K.Anbazhagan, "Design and Implement an Association Rule hiding Algorithm for Privacy Preserving Data Mining" International Journal of Advanced Research in Computer and Communication Engineering Vol. 1, Issue 7, September 2012.

[2] M. Atallah, E. Bertino, A. Elmagamind, M. Ibrahim, and V. S. Verykios “Disclosure limitation of sensitive rules,” .In Proc. of the 1999 IEEE Knowledge and Data Engineering Exchange Workshop (KDEX 1999), pp. 45-52.

[3] Verykios, V.S., Elmagarmid, A., Bertino, E., Saygin, Y., and Dasseni, E. “Association rule hiding”, *IEEE Transactions on Knowledge and Data Engineering*, 2004, 16(4): pp. 434-447.

[4] Y. Saygin, V. Verykios, and C. Clifton, “Using Unknowns to Prevent Discovery of Association Rules” ACM SIGMOD, Vol. 30, No. 4, pp. 45–54, 2001.

[5] Y. Saygin, V. Verykios, and A. Elmagarmid, “Privacy preserving association rule mining,” In: Proc. Int’l. Workshop on Research Issues in Data Engineering (RIDE 2002), pp.151–163, 2002.

[6] Marzena Kryszkiewicz. “Representative Association Rules”, In proceedings of PAKDD’98, Melbourne,Australia(Lecture notes in artificial Intelligence,LANI 1394, Springer-Verleg, pp 198-209, (1998)

[7] A. Jafari and S-L. Wang, “Using unknowns for hiding sensitive predictive association rules,” In IEEE International Conference on Information Reuse and Integration, pp. 223 – 228, (2005)

[8] H. Mannila and H. Toivonen, “Levelwise search and borders of theories in knowledge discovery” *Data Mining and Knowledge Discovery*, vol.1 (3), Sep. 1997, pp. 241-258.

[9] Gkoulalas-Divanis and V.S. Verykios, “An Integer Programming Approach for Frequent Itemset Hiding”, In Proc. *ACM Conf. Information and Knowledge Management (CIKM ’06)*, Nov. 2006.

[10] Gkoulalas-Divanis and V.S. Verykios, “Exact Knowledge Hiding through Database Extension,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 21(5), May 2009, pp. 699-713.