An Enhanced Association Rule Mining To Find Sensitive Patterns And Hide Them For Privacy Preservation

Chandrima R Ghosh[1] ,Jasmine Jha[2]
[1]Student,Department of Computer Engineering,L. J. I. E. T, Ahmedabad, Gujarat, India
[2]Assistant Professor,Department of Computer Engineering,L. J. I. E. T, Ahmedabad, Gujarat, India

Abstract

As the data is doubling every minute, we have propose a RHID algorithm (Rule hiding for incremental datasets) that work with the issue of incremental dataset.It reduces the burden of Incremental data. It works in two parts MDSRRC techniques santised the original data and creates a template table with the use of it hides sensitive patterns in incremental data. Experiment results show that the proposed method can efficiently hide sensitive rules in the incremental environment.

Keywords—*Association Rule Hiding , Privacy Preservation Data Mining , Sensitive Patterns , RHID*

.1.INTRODUCTION

Data Mining is the process of extracting useful knowledge from large amounts of data. Data should be manipulated in such a sensitive way that information cannot be found through Data Mining techniques .While handling sensitive information it becomes very important to protect data against unauthorized access. This has increased the disclosure risks when the data is released to outside parties. This scenario leads to the research of sensitive knowledge hiding in database. [1]

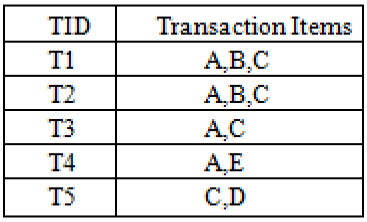
1.1 Privacy Preserving Data Mining (PPDM)

It is considered to maintain the privacy of data and knowledge extracted from data mining. It allows the protection of sensitive data or information while extracting. To preserve data privacy in terms of knowledge, one may modify the original database in such a way that the sensitive knowledge is not involved the mining result and non sensitive knowledge will be extracted. In order to protect the sensitive association rules, privacy preserving data mining include the area called "association rule hiding" [1]

1.2 Association Rule Mining

Association rule mining is the most effective data mining technique to discover hidden pattern from large volume of data. It was first introduced by R. Agarwal [12] in 1993. It works as follows: Suppose I = {i1, i2, ... , im } as a set of items, D = {t1, t2 , ... , tn} be a set of transactions where ti □ I. A unique identifier, TID, is associated with each transaction. A transaction t supports X, where a one set of items named as I, if X □ t. For example, let take a sample database of transactions.[1]

Table 1.Sample Transaction Table [1]



TID	Transaction Items
T1	A,B,C
T2	A,B,C
T3	A,C
T4	A,E
T5	C,D

An association rule is in the form X => Y, where X and Y are the subsets of item set in I, X □ I, Y □ I, and X∩Y=Ø. In the rule X => Y, where X is called the antecedent (left-hand-side) and Y is the consequent (right-hand-side). Association rule mining generates high number of rules and only few of them are of interest. To solve the interested measurement problem, minimum support and minimum confidence thresholds are applied to each rule: Support for a rule X => Y, is denoted by S(X=>Y), is the proportion of transaction in the data set which contain the item set and is defined as[1]:

$$\text{Support}(X{=>}Y) = |X \cap Y| \ / \ |D|,$$

Where |X∩Y| is the number of transaction containing the item set X and Y in the database, |D| denotes the number of the transactions in the data.

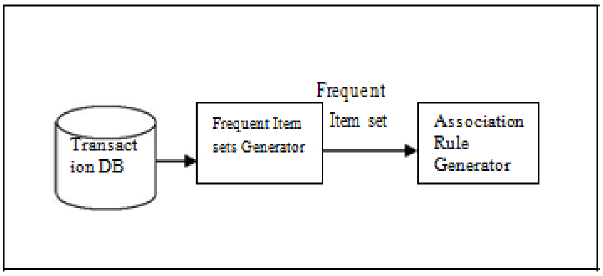
Confidence for a rule X => Y, is denoted by C (X =>Y), is taken as ratio of the support count of X union Y to that of the antecedent X defined as[1] :

$$\text{Confidence}(X{=>}Y) = |X \cap Y| \ / \ |X|,$$

Where |X| denotes the number of the transactions in the database D that contains item set X. In other words, support describes how often the rule would appear in the database, while confidence measures the strength of the rule. A rule
X=>Y is strong if support(X=>Y) ≥ minimum support and Confidence (X=>Y) ≥ minimum confidence

i. First find all frequent item sets- item set which occur at least as frequently as pre-determined minimum support count.[1]
ii. Generate stronger association rules which are based on the user defined minimum support and minimum confidence.[1]

Figure 1. Association Rule Mining Process[1]



There are different types of association rule mining algorithms which are available like Apriori algorithm, Partition algorithm, Dynamic item set counting algorithm, FP tree growth algorithm, etc. Apriori algorithm is one of the most popular and best-known algorithm to mine association rule, proposed by Agrawal and Srikant [1]. It makes user of prior knowledge of frequent itemset properties, which is a two-step process: join step and prune step. It moves upward in the lattice starting from level1 till level k and in reult there is no candidate set will remains after pruning.

1.3 Association Rule Hiding for PPDM

Association Rule hiding is the process of hiding strong association rules and creating sanitized database from the original database in order to prevent unauthorized party to generating frequent sensitive patterns. The problem can stated as " Given a transactional database D , minimum confidence as well as minimum support and a set R of rules can be mined from database D." A subset of R is denoted as set of sensitive association rules which Are to be hidden. The objective is to transform D into a database D' in such a way that no association rule in subset will be mined and all non sensitive rules in R could still be mined from D'

A rule for example X => Y, can be done by two ways it can be either by decreasing the support of the item set X and Y can below the minimum support threshold or decreasing the confidence of the item set X and Y can below minimum confidence threshold. Decreasing the confidence of a rule X => Y can be done by either increasing the support of X in transactions and not of Y or by decreasing the support of Y in transactions supporting both XY. Decreasing the support of a rule X => Y can be done by decreasing the Support of the corresponding large item set XY. [1]

Association rule hiding must satisfy some conditions which are given below:

i. Sensitive rule should not be generated from database.[1]
ii. Non sensitive rule must be generated from Sanitized database.[1]
iii. No new rule which is present in database should be generated from Sanitized database.[1]

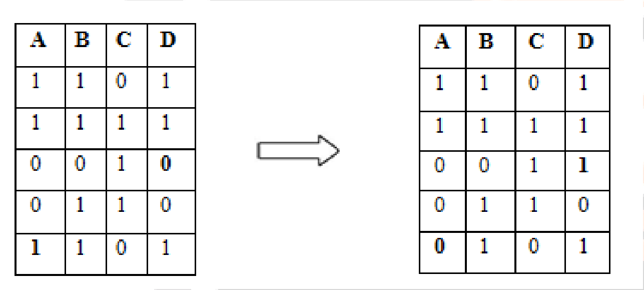
1.4. Association Rule Hiding Approaches

Association Rule Hiding approaches can be classified into five classes which is discussed below: [1]

1.4.1 Heuristic Based Approaches is further divided into two techniques: i) Data distortion technique and ii) Data Blocking Technique.

A. Data distortion technique:- In this technique we replace 1- values to 0-values (delete items) or 0-values to 1-values(add items). There two approaches for rule hiding in data distortion based technique. First is reducing the confidence of rules and second is reducing the support of rules.

Table 2. Hiding A=>D by Distortion[1]

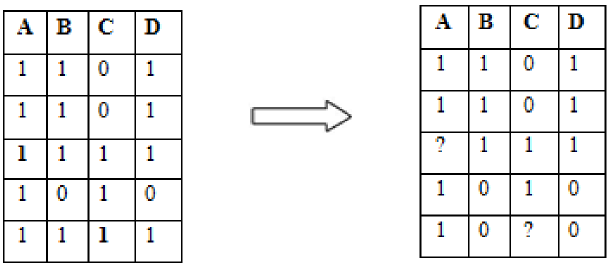


A	B	C	D
1	1	0	1
1	1	1	1
0	0	1	0
0	1	1	0
1	1	0	1

A	B	C	D
1	1	0	1
1	1	1	1
0	0	1	1
0	1	1	0
0	1	0	1

B. Data Blocking Technique is used to increase or decrease the support of the items by replacing 0's or 1's by unknowns "?", so that it become difficult for an adversary to know the value behind "?". This technique is effective and provides certain privacy. When hiding many of the rules at one time then they require less number of database scans and prune more number of rules.

Table 3. Hiding A=>C by Blocking[1]



A	B	C	D
1	1	0	1
1	1	0	1
1	1	1	1
1	0	1	0
1	1	1	1

A	B	C	D
1	1	0	1
1	1	0	1
?	1	1	1
1	0	1	0
1	0	?	0

1.4.2 Border Based Approach

Its hides sensitive association rule by modifying the borders in the lattice of the frequent and the infrequent item sets 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. It uses the border of non-sensitive frequent item and computes the positive and negative borders in the item set. [1]

1.4.3 Exact Approach

It contains non heuristic algorithms which formulates the hiding process as a constraints satisfaction problem or an optimization problem which is solved by integer programming .In this approach minimally extends the original database by a synthetically generated database called extended database and formulates the construction of the extended database as a constraint satisfaction problem (CSP) which is then solved by using Binary Integer Programming (BIP) and the solution for association rule hiding is nothing but determining a sanitized database by satisfying constraints. [1]

1.4.4 Reconstruction Based Approach

It is implemented by perturbing the data first and reconstructing the distributions at an aggregate level in order to perform the association rules mining. In which first places the original data aside and start from knowledge base. This approach has three phases in which first phase can generates frequent item sets from the original database and second phase performs sanitization algorithm over frequent item sets by selecting hiding strategy and identifying sensitive frequent items sets according to sensitive association rules. The third phase generates sanitized database by using inverse frequent item set mining algorithm and then releases this database. [1]

1.4.5 Cryptography Based Approach

It is used for multiparty computation, when database is distributed among several sites. Multiple number parties may share their private data without leaking any sensitive information at their end. It is divided into two categories: vertically partitioned distributed data and horizontally partitioned distributed data. In these approaches instead of distorting the database, it encrypts original database itself for sharing. The communication cost of this approach is very effective. [1]

2. RELATED WORK

Domadiya (2013)[4] described a heuristic based algorithm for hiding sensitive association rule, the algorithm named as Modified Decrease Support of R.H.S. item of Rule Clusters (MDSRRC) which is basically the algorithm is the modification of algorithm DSRRC and overcome the limitation of DSRRC, it is able to hide the sensitive association rule that contain multiple items in right hand side. The main advantage of proposed algorithm is, it does not make major changes in the database and it also able to hide rule which contain multiples item in right hand side of the rule.

Jadav (2013)[5] describes that database containing sensitive knowledge must be protected against unauthorized access. It has become necessary to hide sensitive knowledge in database. To address this problem, Privacy Preservation Data Mining (PPDM) include association rule hiding method to protect privacy of sensitive data against association rule mining. Various existing approaches to association rule hiding have been surveyed along with some open challenges.

Modi (2013)[2] proposed a heuristic algorithm named Decrease Support of R.H.S. item of Rule Clusters in short DSRRC ,which was able to hide many sensitive association rule at a time. They have analyzed experimental results for DSRRC, which show that performance of the DSRRC algorithm is better than other existing heuristic approaches. They have achieved improvement in misses cost, art factual patterns, dissimilarity and maintain data quality. This approach was able to hide only those rules that come on the right hand side (R.H.S.) contain single item, of the rule.

Weng[3]proposed an efficient algorithm, FHSAR which is used for fast hiding sensitive association rules. The algorithm can completely hide any given SAR by scanning database only one time and which significantly reduces the time while execution. Experimental results also show that FHSAR performs better than previous works in terms of execution time required and side effects which are generated in most cases.

Pathak (2012)[11] proposed an approach that is based on concept of pc cluster which improve performance by running operations in parallel, impact factor of a transaction which is equal to number of item sets that are present in those item sets which represents sensitive association rule and hybrid algorithm which is a combination of ISL (Increase support of LHS) and DSR (Decrease support of RHS). This approach is able to reduce the execution time and maintain data quality.

Sharma et al.(2014)[8] proposed an approach that is based
on the combined techniques of randomization and k-anonymization. is divded into two algorithms. In algorithm I randomization is performed on dataset using attribute transitional probability matrix and in algorithm II k-anonymity is performed on randomized dataset which is result of algorithm I.

3. PROBLEM DESCRIPTION

The association rule hiding problem is to sanitize database in a way that through association rule mining one will not be able to infer the specified sensitive rules and will be able to mine all the non-sensitive rules. let us given dataset D, a set of association rules R mined from D and also R0□R, R0 is specified a set of sensitive rules which is to be hidden. The problem is to find sanitized database D' such that there exists only a set of rules R-R0. Now suppose, we have incremental dataset D1 then our objective is to find sanitized database D" such that D+D1 should not contain R0.

4. PROPOSED WORK

4.1 Proposed System Flow-Diagram

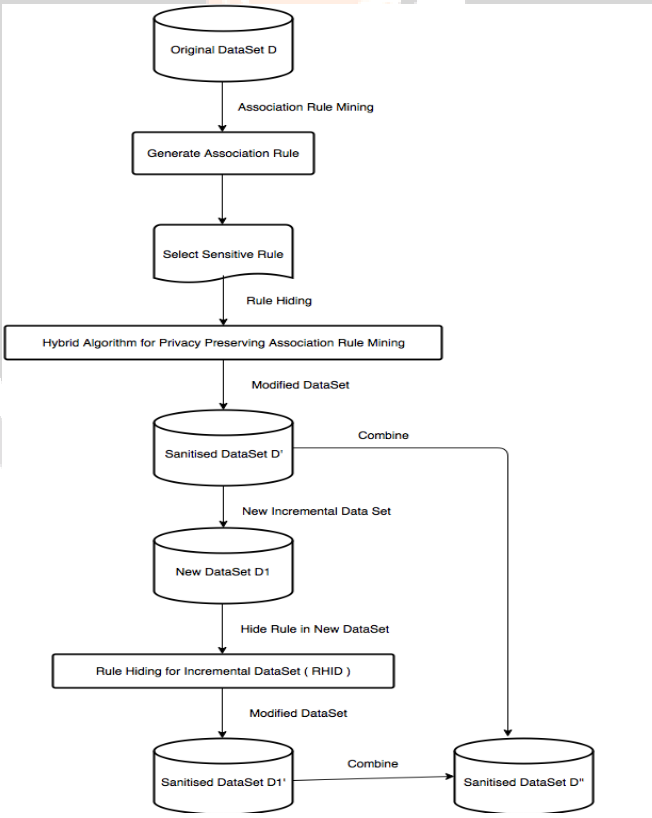


Figure 2: Proposed System Flow Diagram

The framework can be divided into two parts:- The first is rule hiding in original database that uses Hybrid(ISL and DSR) algorithm and the second is rule hiding in incremental database for which we will use our proposed RHID (Rule Hiding for Incremental Dataset) Algorithm.

The framework will work. First we will mine the association rules AR from original database D using Apriori Algorithm, then from this user will manually select the sensitive association rules SR. After selection of sensitive association rules SR we will apply Hybrid(ISL and DSR)Algorithm for hiding purpose from original database D. Once the hiding is done in original database D sanitized database D' is produced. From this sanitized database D' we will generate template table which will help us for hiding the rules in the newly added database D1 for the hiding the sensitive rules. Now for the second part, hiding is done using template table T by choosing the templates and calculating the support count of these templates from the new database D1. If the support count > MST (Minimum Support Threshold) then we will delete this template from new database with the help of RHID algorithm. This process is repeated till support court of templates is less then the defined MST. After this hiding process we get new sanitized database D1'. Then finally combine new sanitized database D1' and sanitized database D' to get the final Sanitized Database D", hence this sanitized database D" can be realized to the outside world. In this way n numbers of incremental files can be added to this system and can generate the final Sanitized Database D" which will be the combination of all Sanitized Database (D1'+ D2'+ D3'........ Dn' = D").

4.2 Proposed Algorithm

INPUT:

Original Database, Required Confident and Support, Sensitive Rules SR

OUTPUT:

Sanitized Database D" with all sensitive rules hidden.

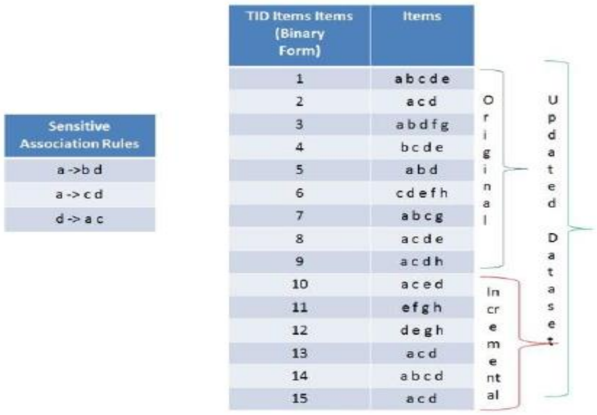
ALGORITHM:

1. Generate Association Rule from Original Database D
2. Select the Sensitive Rules.
3. Now apply MDSRRC Algorithm for Rule Hiding and sanitised dataset will be generated and template table is created
4. Modified Dataset D' is obtained.
5. New Incremental Dataset D1 is added.
6. Rule Hiding for incremental dataset is applied(RHID).
7. Sensitive rules which are manually selected in template table and new datasets 's newly generated sensitive rules are hidden by RHID algorithm.
8. Generate Sanitized Database D1'
9. Combine D1' and D'
10. Sanitized Database D" is generated which can be released.

5. Example

Lets us understand through example, In Table 4 new incremental database D1 is shown which will be added to original database D

Table 4- Sensitive rules and Transaction sets



Sensitive Association Rules
a ->b d
a -> c d
d -> a c

TID Items Items (Binary Form)	Items
1	a b c d e
2	a c d
3	a b d f g
4	b c d e
5	a b d
6	c d e f h
7	a b c g
8	a c d e
9	a c d h
10	a c e d
11	e f g h
12	d e g h
13	a c d
14	a b c d
15	a c d

Original
Incremental
Updated Dataset

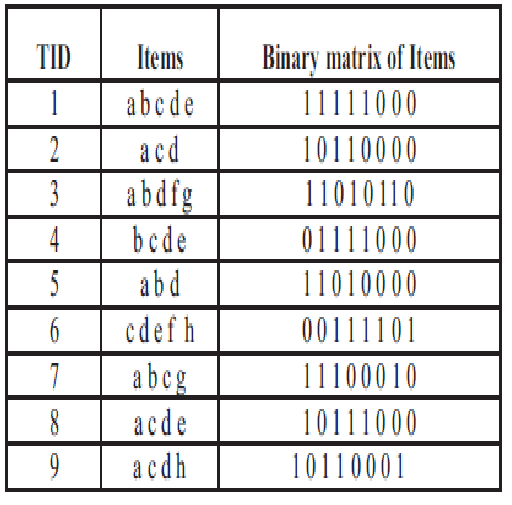
Table 4 shows the original dataset and incremental which is added and chosen sensitive rules.

First by MDSRRC technique original data is sanitized , The sensitivity of a=3, b=1, c=2, d=3. Transaction with its sensitivity is shown in Table III. Now algorithm finds frequency of each item presents in R.H.S of sensitive rules. here frequency of d=2, c=2, a=1, b=1. So IS= {d,c,a,b}. In this example item'd' is selected as is0. Then it sorts the transactions which supports is0 in descending order of their sensitivity.

Then Select transaction with highest sensitivity and delete is0 item from that transaction. Update confidence and support of all the sensitive rules. Table IV show modified database D1 after first deletion of item from first transaction.

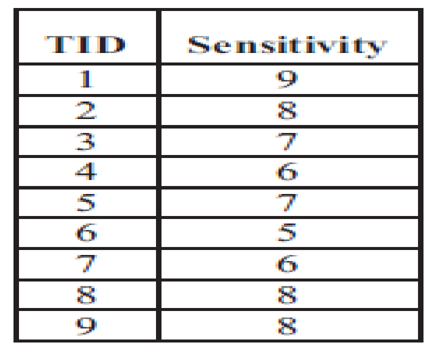
Now update sensitivity of each item. Updated count of each
item for IS is c=2, d=1, a=1, b=1. So updated IS={c,d,a,b} and is0 ='c'. Sort transactions which support is0 and delete the is0 from transaction with highest sensitivity, then delete the is0.

Table 5:- Transactional Data



TID	Items	Binary matrix of Items
1	abcde	11111000
2	acd	10110000
3	abdfg	11010110
4	bcde	01111000
5	abd	11010000
6	cdef h	00111101
7	abcg	11100010
8	acde	10111000
9	acdh	10110001

Table 6:-Transaction with Sensitivity

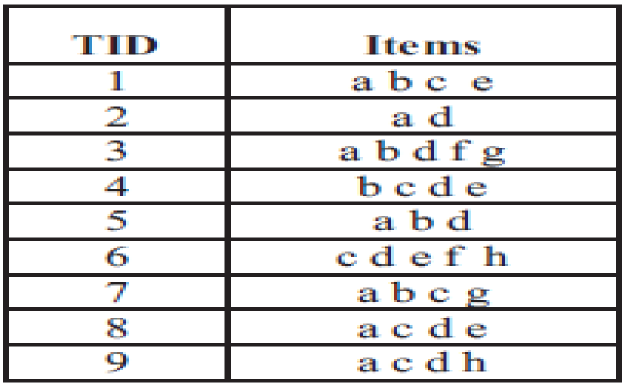


TID	Sensitivity
1	9
2	8
3	7
4	6
5	7
6	5
7	6
8	8
9	8

In this way original data is santised ,now the second part of the algorithm. The frequent items from template table T based on owner specify rule a→bd, a→cd and d→ac as sensitive rules are abd, acd and dac.

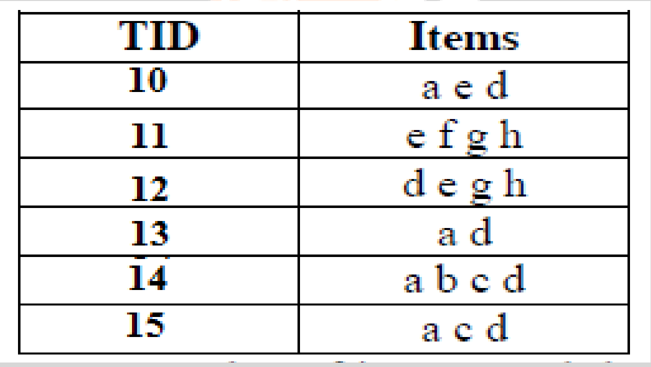
Now we will scan these frequent items from the new dataset D1 to check that whether this frequent items abd, acd and dac are present in the new dataset D1. If present then increment the count with the count in template table T for each of the frequent items. After scanning we get that the count of abd=3, acd=6, dac=6. After that, we will check whether count(frequent item) > MST, if the count is greater than Minimum Support Threshold then sort these items. So after sorting, their order will be {acd=6, dac=6, abd=3}. So as per our example, for acd=6 we will delete item c from transaction1 from D1 which is having all items a, c and d. After that we will again count for this frequent items and check whether count (frequent item) > MST, if count is greater than Minimum Support Threshold then again sort these items. So now {acd=5, dac=5, abd=3} and to hide acd we will delete c transaction 4 as count(acd=5) >MST. This process will continue till count (frequent item) < MST for all frequent items of sensitive rules. At last finally sanitized table D1' for the example will be same as in Table 3. And the final Sanitized database D" which is form by merging sanitized table D' of original Dataset D with sanitized table D1' of incremental Dataset D1.

Table 7:- Sanitised Original Datasets



TID	Items
1	a b c e
2	a d
3	a b d f g
4	b c d e
5	a b d
6	c d e f h
7	a b c g
8	a c d e
9	a c d h

Table 8:- Sanitised Incremental Datasets



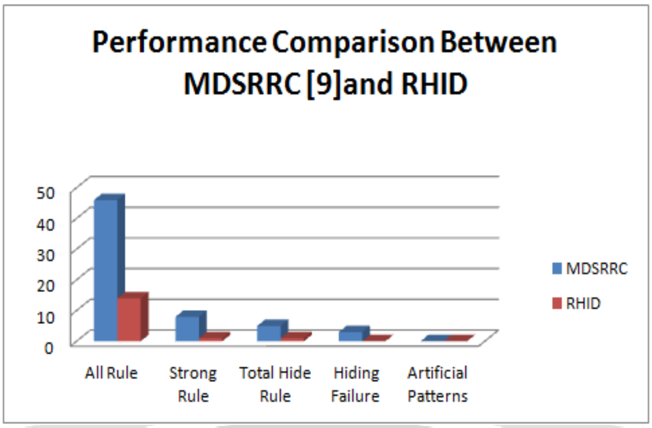
TID	Items
10	a e d
11	e f g h
12	d e g h
13	a d
14	a b c d
15	a c d

6. Experimental Result and Analysis

Table 9:- Performance Result

Parameters	MDSRRC	RHID
All Rules	46	14
Strong Rules	8	1
Hide Rules	5	1
Hiding Failure	3	0
Artificial patterns	0	0

Chart 1 :-Performance Comparison of MDSRRC and RHID



7. CONCLUSION

From performance metrics results we can say that our approach is successfully able to hide the sensitive rules in incremental environment also, for which less time and less computation is required. Thus, it provides certain level of privacy and security for incremental dataset of same behaviour.

REFERENCES

[1]Kenampreet Kaur, Meenakshi Bansal"A Review on various techniques of hiding Association rules in Privacy Preservation Data Mining" IJECS Volume 4 Issue 6 June, 2015 Page No.12947-12951.
[2]C. N. Modi, U. P. Rao, and D. R. Patel ., "Maintaining privacy and data quality in privacy preserving association rule mining," in Second International conference on Computing, Communication and Networking Technologies, pp. 1–6, Jul. 2010.
[3]C.Weng, S. Chen, H. Lo, "A Novel Algorithm for Completely Hiding Sensitive Association Rules," IEEE – 2008 Intelligent Systems Design and Applications, vol 3, pp.202-208.
[4]N. Domadiya and U.P. Rao, "Hiding Sensitive Association Rules to Maintain Privacy and Data Quality in Database," 3rd IEEE International Advance Computing Conference (IACC), pp. 1306-1310, 2013
[5]K.B.Jadav, J. Vania, D R. Patel, "A Survey on Association Rule Hi ding Methods," International Journal of Computer Applications (0975 – 8887), 82 (13), pp-20-25, 2013
[6]Mohnish Patel, Prashant Richhariya, Anurag Shrivastava "A Novel Approach for Data Mining Clustering Technique using NeuralGas Algorithm" IEEE,2014 pp. 251-254
[7]Manish Sharma, Atul Chaudhar,Manish Mathuria , Shalini Chaudhar, Santosh Kumar" An Efficient Approach for Privacy Preserving in Data Mining " IEEE,2014 pp.244-249
[8]Mr.S.Chidambaram , Dr.K.G Srinivasagan, "A Combined Random Noise Perturbation Approach for Multi Level Privacy Preservation in Data Mining" IEEE,2014 pp.1-5
[9] Bhoomika R Mistry, Amish Desai" Privacy preserving heuristic approach For Association Rule Mining in Distributed Database"IEEE.2015 pp.1-6
[10] K. Pathak, N. S. Chaudgari and A. Tiwari, " Privacy Preserving Association Rule Miningby Introducing Concept of Impact Factor," in 7th IEEE Conference on Industrial Electronics and Application(ICIEA), pp. 1458-1461, 2012.
[11] R. Agrawal, T.Imielinski, and A. Swami, R.Srikant, "Mining association rules between sets of items in large databases,"In Proceedings of ACM SIGMOD International Conference on Management of Data, Washington, DC, pp. 207-216, 1993.
[12] Shyue-Liang Wang, Ayat Jafari "Hiding Sensitive Predictive Association Rules"IEEE-2005