International Journal of Computer Theory and Engineering, Vol. 1, No. 4, October, 2009
1793-8201

Privacy Preserving Fuzzy Association Rules Hiding in Quantitative Data

Manoj Gupta and R. C. Joshi

***Abstract*—Data mining is the process of extracting useful patterns or knowledge from large databases. However, data mining also poses a threat to privacy and information protection if not done or used properly. Therefore, researchers need to investigate data mining algorithm from a new point of view that is of personal privacy. Many algorithms have been developed to hide association rules discovered from a binary database. But in real applications, data mostly consists of quantitative values. In this paper, we thus propose a fuzzy association rules hiding algorithm for hiding rules discovered from a quantitative database. The proposed algorithm integrates the fuzzy set concepts and Apriori mining algorithm to find useful fuzzy association rules and then hide them using privacy preserving technique. For hiding purpose, we decrease the support of the rule to be hidden by decreasing the support value of item in either Left Hand Side (L.H.S.) or Right Hand Side (R.H.S) of the rule. Experimental results show that the proposed algorithm hides more rules and maintains higher data quality of the released database.**

***Index Terms*— Fuzzy association rules, fuzzy set concepts, privacy preserving data mining, quantitative data.**

I. INTRODUCTION

Data mining is the process of extracting useful patterns or knowledge from large databases. However, data mining also poses a threat to privacy and information protection if not done or used properly. For example, association rule analysis is a popular tool for discovering useful associations from large amount of data and some useful hidden information could be easily discovered using this kind of tool. Therefore, the protection of sensitive hidden information has become a critical issue to be resolved.

The objective of privacy preserving data mining is to hide certain information so that they cannot be discovered through data mining techniques such as association rule analysis [1]. There have been two broad approaches for privacy preserving data mining [2]-[5]. The first approach, called output privacy, is to alter the data before delivery to data miner so that real data is obscured and mining result will not disclose certain privacy. For example, perturbation, blocking, merging, swapping and sampling are some methods that have been proposed for this type of output privacy [6]. The second approach, called input privacy, is to manipulate the data using data distribution methods. In this approach, mining result is not affected or minimally affected. For example, reconstruction based and cryptography based are some techniques that have been proposed for this type of input privacy [6].

Wang et al. [3]-[5] proposed two algorithms namely ISL (Increase Support of Left hand side) and DSR (Decrease Support of Right hand side) to hide useful association rule from transactions data with binary attributes. In ISL method, confidence of a rule is decreased by increasing the support value of Left Hand Side (L.H.S.) of the rule. For this purpose, only the items from L.H.S. of a rule are chosen for modification. In DSR method, confidence of a rule is decreased by decreasing the support value of Right Hand Side (R.H.S.) of a rule. For this purpose, only the items from R.H.S. of a rule are chosen for modification.

As mentioned above, almost all of studies proposed concentrated on hiding boolean association rules which are concerned only with whether an item is present in a transaction or not, without considering its quantity. However, transactions with quantitative values are commonly found in real world applications. For example, in a patient's blood test, many attributes could be found. However, attribute's quantity instead of just presence/absence of attribute in blood is more important for determination of illness. For example, many people have the problem of sugar, but this doesn't mean that one is sick or not, the only criterion for determination of illness is the surplus/deficiency in sugar's quantity.

The problem of mining quantitative association rule was first introduced in [7]. The basic idea was to map the categorical attribute values into corresponding binary attribute values. Some work has been done to discover fuzzy association rules from quantitative data using fuzzy set concepts [8]-[11]. However, only one work has been done in the field of hiding fuzzy association rule in quantitative data [12]. T. Berberoglu et al. [12] proposed an algorithm to hide fuzzy association rule in quantitative data. The basic idea of this algorithm was to decrease the confidence of a rule by increasing support of L.H.S. of rule.

In this paper, we attempt to present a method for preventing extraction of useful association rules from quantitative data by decreasing the support of the rule. The support of a rule $A \rightarrow B$ is decreased by decreasing the support count of itemset AB which is achieved by decreasing the support value of either A or B i.e. item in L.H.S. or R.H.S. of the rule and this is done until either support or confidence value of the rule goes below minimum support or minimum confidence value respectively.

The rest of this paper is organized as follows. Privacy preserving fuzzy association rule hiding problem in quantitative data is defined in Section II. Our approach to

Manuscript received May 14, 2009.

F. Manoj Gupta is a student of Master of Technology in Computer Science and Engineering at Indian Institute of Technology Roorkee, Uttarakhand, India. (Phone: +919897598924)

S. R. C. Joshi is a professor with Electronics and Computer Engineering department at Indian Institute of Technology Roorkee, Uttarakhand, India..

hide useful fuzzy association rules is described in Section III. The fuzzy association rules hiding process is presented in Section IV. The proposed algorithm related example is included in Section V. Experimental results are given in Section VI. Analysis is given in Section VII. Section VIII includes the conclusions.

II. Problem Statement

Fuzzy association rule hiding algorithm first finds the useful fuzzy association rules in quantitative data using fuzzy set concepts and then hide them using privacy preserving technique. Mining fuzzy association rules is the discovery of association rules using fuzzy concepts such that the quantitative attributes can be handled properly.

Let $I = \{ i_1, i_2, ..., i_m \}$ be the complete item set where each $i_j \, (1 \le j \le m)$ is a quantitative attribute. Given a database $D = \{ t_1, t_2, ..., t_n \}$ with attributes I and the fuzzy sets associated with attributes in I, we want to find out some interesting useful association rules.

Let $X = \{ x_1, x_2, ..., x_p \}$ and $Y = \{ y_1, y_2, ..., y_q \}$ are two large itemsets. Then, the fuzzy association rule is given as follows:

$$A \rightarrow B$$

where $A = \{ f_1, f_2, ..., f_p \}$ and $B = \{ g_1, g_2, ..., g_q \}$ and

$f_i \in$ {the fuzzy regions related to attribute x_i}

$g_j \in$ {the fuzzy regions related to attribute y_j}

X and Y are subsets of I and are disjoint which means that they share no common attributes. A and B contain the fuzzy sets associated with the corresponding attributes in X and Y. Here, A is called as the body or Left Hand Side (L.H.S.) of the rule and B is called as the head or Right Hand Side (R.H.S.) of the rule. The significance of an association rule is measured by its *support* and *confidence* [13]. Support is defined as the percentage of transactions that contain both A and B, while confidence is defined as the ratio of the support of $A \cup B$ to the support of A. In other words, the support of a rule measures the significance of the correlation between itemsets, while the confidence of a rule measures the degree of the correlation between itemsets. If a rule is *useful/interesting*, it should have support larger than or equal to minimum support value and confidence larger than or equal to minimum confidence value.

III. Proposed Algorithm

In order to hide an association rule, $A \rightarrow B$, we can either decrease its support to be smaller than minimum support value or its confidence to be smaller than its minimum confidence value. To decrease the confidence of a rule, two strategies can be used. The first one is to increase the support count of A i.e. L.H.S. of the rule, but not support count of $A \cup B$. The second one is to decrease the support count of $A \cup B$, while keeping the support count of A i.e. L.H.S. of the rule constant.

Based on first method mentioned above, we proposed an algorithm namely Decrease Rule Support (DRS). This algorithm first finds the useful fuzzy association rules which consist of only one item on both sides of the rule and then hide them using privacy preserving technique. For hiding purpose, the algorithm tries to decrease the support of rule $A \rightarrow B$ by decreasing the support count of itemset AB until either support or confidence value of the rule goes below minimum support or minimum confidence value respectively. To achieve this, the support count of itemset AB is decreased by decreasing the support count of either A or B i.e. item in L.H.S. or R.H.S. of the rule. For this purpose, the value of item in L.H.S. or R.H.S. is subtracted from one in case one minus value of item in L.H.S. or R.H.S. is less than the value of item in R.H.S or L.H.S respectively.

Abbreviations used in the proposed algorithm are given as follows:

D: Initial database with *n* transaction data; F: fuzzified database; T_L: value of a L.H.S. item in transaction t; T_R: value of a R.H.S. item in transaction t; U: An association Rule; R_h: Rules to be hidden.

Input:

(1) A source database D,
(2) A minimum support value (min_support),
(3) A minimum confidence value (min_confidence).

Output:

A transformed database D' so that useful fuzzy association rules cannot be mined.

Algorithm DRS:

```
1.  Fuzzification of the database, D → F;
2.  In fuzzified database F, calculate every item's support
    value where f ε F;
3.  IF all f (support) < min_support THEN
4.    EXIT;   // there isn't any rule
5.  Find large 2-itemsets from F;
6.  FOR EACH X's large 2-itemset {//find all rules
7.    Find R = {Rules from itemset X};
      //for X= {i1, i2}, two possible rules are i1 → i2
      //and i2 → i1
8.    IF R is empty THEN
9.        GO TO next large 2-itemset; //i.e. line 6
10.   Select and remove a rule U from R;
11.   Compute confidence of the rule U;
12.   IF confidence (U) < min_confidence THEN
13.      Add the rule U to Rh;
14.   GO TO line 8;
15. } //end of FOR EACH line 6
      //Hides all rules in Rh
16. REPEAT {//until no more rule can be hidden
17.   Select the next rule U from Rh;
18.   IF confidence (U) < min_confidence OR
      support < min_support THEN
19.     GO TO line 16;
20.   Find Tx = {t| t ε U such that 1-max (TL, TR) <
      min (TL, TR)};
21.   Sort transactions in Tx in descending order by
      value TL + TR -1;
      // for maximum decrease in support value of rule
22.   WHILE (confidence(U) ≥ min_confidence and
      support (U) ≥ min_support and Tx is not empty) {
23.       Choose the first transaction t from Tx;
```


IACSIT
International Association of Computer Science and Information Technology
WWW.IACSIT.ORG

```
24.     IF T_R > 0.5 and T_L = T_R THEN
25.       T_R = 1 – T_R;
26.     ELSE
27.       max (T_L,T_R) = 1 – max (T_L,T_R);
28.     Remove and save the transaction t from Tx;
29.     Re-compute support and confidence of rule U
30.   } // end WHILE line 22
31.   IF Tx is empty THEN
32.     Cannot hide rule U and restore F;
33. } UNTIL (No rule in R_h is modified)//end line 16
34. Transform the updated database F→D' and output
    updated D';
```

IV. STEPS OF FUZZY ASSOCIATION RULE HIDING ALGORITHM

In this section, the fuzzy concepts are being used in the apriori association rules mining algorithm to discover useful fuzzy association rules from quantitative data that are to be hidden using privacy preserving technique. Notation used in paper is stated as follow:

n : the total number of transactions data;
m : the total number of attributes (items);
$D^{(i)}$: the i^{th} transaction data, $1 \le i \le n$;
I_j : the j^{th} attribute, $1 \le j \le m$;
$\left| I_j \right|$: the number of fuzzy regions for I_j ;
R_{jk} : the k^{th} fuzzy region of I_j , $1 \le k \le \left| I_j \right|$;
$v_j^{(i)}$: the quantitative value of I_j for $D^{(i)}$;
$f_{jk}^{(i)}$: the membership value of $v_j^{(i)}$ in the region R_{jk} ;
a : the minimum support value;
l : the minimum confidence value.

The given fuzzy hiding algorithm first transforms each quantitative value into fuzzy sets using membership function. The mining process based on fuzzy counts is then performed to find useful fuzzy association rules that are to be hidden. The detailed steps of the fuzzy association rules hiding algorithm is described as follows.

STEP 1: For each transaction data $D^{(i)}$, i=l to n, and for each attribute (item) I_j , j=l to m, transform the quantitative value $v_j^{(i)}$ into a fuzzy set $f_{jk}^{(i)}$ represented as $\left(\frac{f_{j1}^{(i)}}{R_{j1}} + \frac{f_{j2}^{(i)}}{R_{j2}} + - - - + \frac{f_{jp}^{(i)}}{R_{jp}} \right)$ using the given membership function for the attribute, where p is the number of fuzzy regions for attribute I_j .

STEP 2: Calculate the support count of each attribute region R_{jk} on the transactions data as: $count_{jk} = \sum_{i=1}^{n} f_{jk}^{(i)}$.

STEP 3: For each attribute region R_{jk} , $1 \le j \le m$ and $1 \le k \le \left| I_j \right|$, check whether its $count_{jk}$ is greater than or equal to the given minimum support value α. If R_{jk} satisfies the above condition, then put it in the set of large 1-itemsets (L_1). That is:

$$L_1 = \left\{ R_{jk} \,\middle|\, count_{jk} \ge a , 1 \le j \le m \, and \, 1 \le k \le \left| I_j \right| \right\}$$

STEP 4: Join the large 1-itemsets (L_1) to generate the candidate set C_2 in a way similar to that in apriori algorithm except that two regions belonging to the same attribute (item) cannot simultaneously exist in an itemset in C_2.

STEP 5: For each candidate 2-itemset S with regions (A_1 and B_1) in C_2, do the following steps:

(i) Calculate the fuzzy value of each transaction data on itemset S as:

$$f_S^{(i)} = \underset{j=1}{\overset{2}{Min}} \, f_{S_j}^{(i)}$$

(ii) Calculate the fuzzy count of itemset S on the transactions data as:

$$count_S = \sum_{i=1}^{n} f_S^{(i)}$$

(iii) If $count_S$ is greater than or equal to the given minimum support value α, then put the itemset S in set L_2 (Large 2-itemset).

STEP 6: For each large 2-itemset, find the interesting useful association rules having confidence value greater than or equal to minimum confidence value λ. The confidence value of a rule $A_1 \to B_1$ is computed as follows:

$$Confidence\,(A_1 \to B_1) = \frac{Support\,(A_1 \to B_1)}{Support\,(A_1)}$$

where support of itemset S with items (A_1 and B_1) is computed as follows:

$$Support \;(S) = \frac{Count_S}{N}$$

STEP 7: In order to hide the sensitive rules, support values are tried to be decreased which is achieved by decreasing the support count of itemset AB and this is done until either support or confidence value of the rule goes below minimum support or minimum confidence value respectively. The support count of itemset AB can be decrease by decreasing the support count of either A or B i.e. item in L.H.S. or R.H.S. of the rule. That is:

$$Support \;(A_1 \to B_1) = \frac{Support \;(A_1 B_1)\downarrow}{N} \qquad (1)$$

and

$$Confidence \;(A_1 \to B_1) = \frac{Support \;(A_1 B_1)\downarrow}{Support\,(A_1)} \qquad (2)$$

V. EXAMPLE

In this section, we give an example to demonstrate how the proposed algorithm can be used to hide useful fuzzy association rules from a set of quantitative transaction data [12].

TABLE II. FUZZIFICATION OF TRANSACTION DATA

Transaction↓		**A**			**B**			**C**			**D**	
Regions→	A_1	A_2	A_3	B_1	B_2	B_3	C_1	C_2	C_3	D_1	D_2	D_3

T_1	0	1	0	1	0	0	0.4	0.6	0	0.6	0	0
T_2	0.6	0	0	0	0.8	0.2	0.8	0.2	0	0	0.2	0.8
T_3	0.8	0.2	0	0.6	0	0	0.2	0.8	0	0	0.4	0.6
T_4	0.6	0.4	0	1	0	0	0.4	0.6	0	0	0.6	0.4
T_5	0	0.8	0.2	0.8	0	0	0.6	0.4	0	0	1	0
Count	2.0	**2.4**	0.2	**3.4**	0.8	0.2	**2.4**	**2.6**	0	0.6	**2.2**	1.8

TABLE I. THE SET OF 5 QUANTITATIVE TRANSACTION DATA [12]

	A	B	C	D
T_1	10	5	8	3
T_2	3	11	6	14
T_3	6	3	9	13
T_4	7	5	8	12
T_5	11	4	7	10

In this example, each attribute (item) has three fuzzy regions namely: 1, 2 and 3. Therefore, three fuzzy membership values are produced for each item according to the predefined membership function as shown in Fig. I. For simplicity, we use a single membership function for all the items. Note that different membership functions can be used for different items. Also, we assume that minimum support count and minimum confidence value are set at 2.2 (44%) and 75%, respectively. Fuzzification of transaction data in Table I is given in Table II.

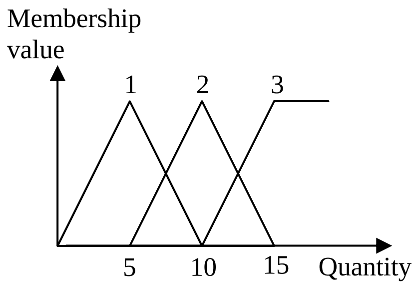

Fig. I. The Membership Function used in this Example

After applying the algorithm, we get the rule $A_2 \rightarrow B_1$ (Support = 2.4 (48%) and Confidence = 100%). In order to hide this rule, we need to decrease its support value. For this purpose, we decrease the value of support (A_2B_1) by decreasing the support of item A_2 or B_1.

Support and Confidence value of the rule $A_2 \rightarrow B_1$ are calculated as:

$$Support\ (A_2 \rightarrow B_1) = \frac{Support\,(A_2 B_1)}{N} = \frac{2.4}{5} = 48\%$$

and

$$Confidence\ (A_2 \rightarrow B_1) = \frac{Support\,(A_2 B_1)}{Support\,(A_2)} = \frac{2.4}{2.4} = 100\%$$

Our Approach:

Firstly, the transactions in Tx are sorted in descending order by value $T_L + T_R - 1$. The transaction set Tx is as shown Table IV. Table III shows transactions data for regions A_2 and B_1.

TABLE III. FUZZY VALUES OF REGIONS A_2 AND B_1

	A_2	**B_1**	**Support**
T_1	1	1	**1**
T_2	0	0	**0**
T_3	0.2	0.6	**0.2**
T_4	0.4	1	**0.4**
T_5	0.8	0.8	**0.8**
Count	**2.4**		**2.4**

TABLE IV. TRANSACTIONS IN TX SORTED IN DESCENDING ORDER BY VALUE $T_L + T_R - 1$

	A_2	**B_1**	**Support**
T_1	1	1	**1**
T_5	0.8	0.8	**0.8**
T_4	0.4	1	**0.4**

To hide rule $A_2 \rightarrow B_1$, the value in transaction T_1 for B_1 is modified from 1 to 1-0 = 1. The new transactions data for regions A_2 and B_1 is as shown in Table V.

TABLE V. TRANSACTIONS DATA FOR REGIONS A_2 AND B_1 AFTER MODIFYING T_1

	A_2	**B_1**	**Support**
T_1	1	**0**	**0**
T_2	0	0	**0**
T_3	0.2	0.6	**0.2**
T_4	0.4	1	**0.4**
T_5	0.8	0.8	**0.8**
Count	**2.4**		**1.4**

Now, support and confidence value of the rule $A_2 \rightarrow B_1$ are calculated as:

$$Support\ (A_2 \rightarrow B_1) = \frac{Support\,(A_2 B_1)}{N} = \frac{1.4}{5} = 28\%$$

and

$$Confidence\,(A_2 \rightarrow B_1) = \frac{Support\,(A_2 B_1)}{Support\,(A_2)} = \frac{1.4}{2.4} = 58.33\%$$

Since the value of support $(A_2 \rightarrow B_1)$ and confidence $(A_2 \rightarrow B_1)$ is less than min_support and min_confidence value respectively, the rule is hidden from the user and we stop our process of modifying data here. Note that if either of support or confidence value goes below minimum support or minimum confidence value respectively, the rule is hidden from user.

Finally, according to new database with modified values, the transformed database is shown in Table VI.

TABLE VI. TRANSFORMED DATABASE D'

	A	B	C	D
T_1	10	**0**	8	3
T_2	3	11	6	14
T_3	6	3	9	13
T_4	7	5	8	12

IACSIT
International Association of
Computer Science and Information Technology
WWW.IACSIT.ORG

T_5	11	4	7	10

VI. Experimental Results

In this section, we present some experiments that have been performed to assess the performance of the proposed algorithm. The performance of the algorithm has been measured according to three criteria: Number of rules hidden, database effects, and side effects produced. As number of rules hidden, we considered the total number of rules hidden for different values of support and confidence. As database effects, we considered the total number of entries modified for hiding a set of randomly selected five rules for different dataset sizes. As side effects, we considered the number of "lost rules" and the number of "new rules" generated by the hiding process. All those rules that can be mined from the source database but cannot be mined from the released database are known as "lost rules". Similarly, all those rules that cannot be mined from the source database but can be mined from the released database are known as "new rules". To determine lost rules, we compared each rule mined from the source database with each rule mined from the released database. If the rule was not found, we considered it as "lost rule." Note that rules selected for hiding are excluded from the set of "lost rules" as they are hidden purposely. To determine new rules, we computed the difference between the number of rules mined from the released database and the number of rules mined from the source database and added to this difference the number of lost rules and the number of rules hidden.

The dataset used is Wisconsin Breast Cancer dataset from UCI Machine Learning Repository [14]. The dataset consists of nine quantitative attributes and one categorical attribute. We used only nine quantitative attributes and ignored categorical attribute. The membership function as shown in Fig. II is used for converting all the nine attributes to their corresponding fuzzy sets.

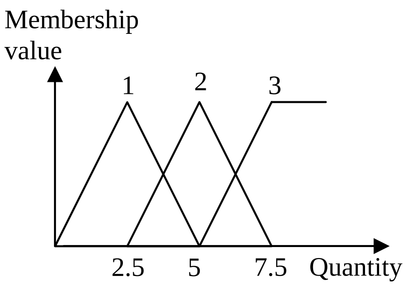

Fig.2 The Membership Function used for the Dataset

We performed six different experiments to compare the performance of the proposed algorithm with previous work [12]. The first experiment finds the relationship between number of total and hidden rules, and number of transactions. In this experiment, the minimum confidence value is set at 70% and minimum support values are taken as 17, 30, 62 and 74 for 50, 100, 150 and 200 transactions respectively. The results are depicted in Fig. III.

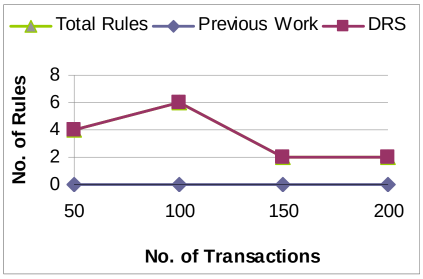

Fig.3. Number of Total and Hidden Rules

The second experiment finds the number of total and hidden rules for different values of minimum support for 200 transactions. The minimum confidence value is set at 40%. The results are depicted in Fig. IV.

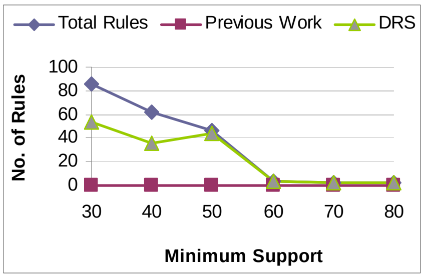

Fig. 4 The Number of Rules under Different Values of Minimum Support

The third experiment finds the number of total and hidden rules for different values of minimum confidence for 200 transactions. The minimum support value is set at 40. The results are depicted in Fig. V.

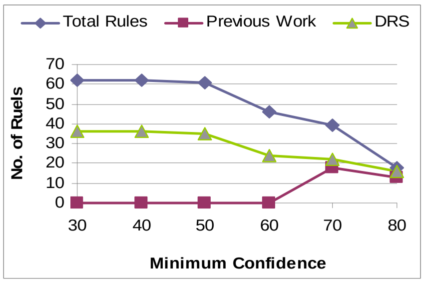

Fig.5 The Number of Rules under Different Values of Minimum Confidence

In next three experiments, the minimum confidence value is set at 74% and minimum support values are taken as 12, 24, 36 and 49 for 50, 100, 150 and 200 transactions respectively. Then, we randomly selected five rules and hide them to find the number of "lost rules", the number of "new rules" generated and total number of entries modified.

The fourth experiment finds the relationship between number of new rules generated as a side effect by hiding process and number of transactions. The results are depicted in Fig. VI.

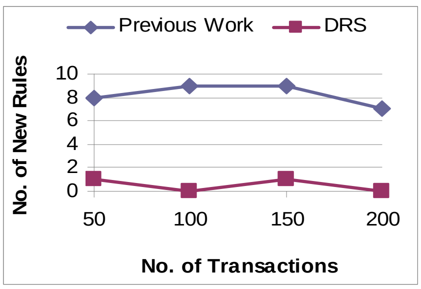

Fig.6 New rules Generated for Hiding a Set of Five Rules

The fifth experiment finds the relationship between number of lost rules and number of transactions. The results are depicted in Fig. VII.

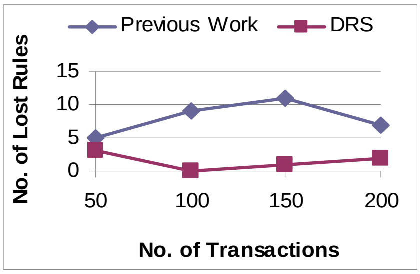

Fig.7 Rules lost after hiding a set of five rules

The final experiment finds the relationship between total number of entries modified and number of transactions. The results are depicted in Fig. VIII.

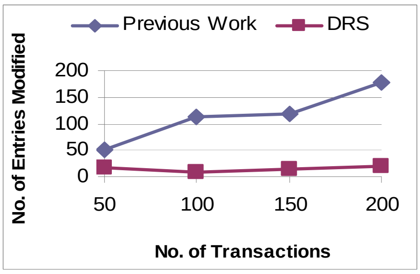

Fig.8 Number of entries modified for hiding a set of five rules

VII. Analysis

This section analyzes some of the characteristics of the proposed algorithm based on our experimental results and compares with the previous work [12]. The first characteristic we observe is the total number of rules hidden for different values of support and confidence. Fig. III shows the relationship between number of total and hidden rules, and number of transactions. It can be clearly seen that our algorithm hides all the rules while previous work hides no rules for this experiment. Fig. IV shows the relationship between the number of total and hidden rules for different values of minimum support. The number of rules decreases rapidly with increase in minimum support value because the number of large 2-itemsets decreases rapidly with increase in minimum support value. Fig. V shows the relationship between the number of total and hidden rules for different values of minimum confidence. The number of hidden rules increases rapidly with increase in minimum confidence value because only a few transactions need to be modified to lower the confidence of the rule for higher minimum confidence value.

From first three experiment results, it can be easily seen that our algorithm hides more rules in comparison to previous work for different value of minimum support and minimum confidence value. The reason is that in our algorithm, a rule $A \rightarrow B$ is hidden by decreasing the support value of $A \cup B$ (see equation 1). In contrast, in previous work, a rule $A \rightarrow B$ is hidden by increasing the support count of A i.e. L.H.S. of the rule, but the support count of $A \cup B$ may also get increased which result in less decrease in the confidence value of the rule. Also, the condition used by previous work allows only a small number of transactions to be modified for the rule under hidden. Therefore, our algorithm hides more number of rules in comparison to previous work.

The second characteristic we observe is the side effects for different dataset of transactions. Fig. VI shows the relationship between number of new rules generated and number of transactions. The number of new rules generated is almost same for all the datasets. Also, our algorithm generates less number of “new rules” in comparison to previous work for all the datasets. Fig. VII shows the relationship between number of lost rules and number of transactions. Our algorithm results in less number of “lost rules” in comparison to previous work for all the datasets.

The last characteristic we observe is the database effects. Fig. VIII shows the relationship between total number of entries modified and number of transactions. Our algorithm modifies a few numbers of entries for hiding a given set of rules in all the datasets.

From last three experiment results, it can be easily seen that our algorithm generate less side effects and modify only a small number of entries in comparison to previous work. The reason is that our algorithm makes minimum modification of data because we used a criterion for selection of transaction for modification. The criterion used is to select transaction in the order which results in maximum decrease in support value of the rule. Therefore, higher data quality of the released database is maintained by our algorithm. In contrast, previous work does not use any criteria for selection of transaction and modifies all possible transactions of a rule and, thus, generate more side effects and reduces the data quality of the released database.

VIII. Conclusions

In this paper, we proposed a hiding algorithm that integrates the fuzzy set concepts and Apriori mining algorithm to find useful fuzzy association rules from a quantitative database and then hide them using privacy preserving technique. Unlike previous approaches which mainly deals with association rules in binary database, our approach deals with hiding the association rules in quantitative database.

Numerical experiments have been performed to measure the performance of the algorithm according to three criteria: the number of rules hidden, side effects and database effects of the algorithm. As from results, we see that our approach is better in a way that it hides more rules for different values of support and confidence. Further, our algorithm makes minimum modification of data and, thus, generates minimal side effects. Therefore, higher data quality of the released database is maintained by our algorithm.

In our algorithm, we consider fuzzy association rules which consist of only one item on both side of the rule. In future, we plan to extend our algorithm to deal with fuzzy association rules which may consist of more than one item on both side of the rule.

IACSIT
International Association of
Computer Science and Information Technology
WWW.IACSIT.ORG

REFERENCES

[1] D.E.O' Leary," Knowledge Discovery as a Threat to Database Security", Proceedings of first International Conference Knowledge Discovery and Databases, (1991, pp. 507-516.

[2] Wang, S.L., Jafari, A., "Using Unknown for Hiding Sensitive Predictive Association Rules", In Proceedings of the 2005 IEEE International Conference on Information Reuse and Integration, USA, August 2005, pp. 223-228.

[3] S. L. Wang, B. Parikh and A. Jafari, "Hiding Informative Association Rule Sets", Expert Systems with Applications, Volume 33, Issue 2, August 2007, pp. 316-323.

[4] S. L. Wang, Y. Lee, S. Billis and A. Jafari, "Hiding Sensitive items in Privacy Preserving Association Rule Mining", IEEE International Conference on Systems, Man and Cybernetics, Vol. 4, Oct 2004, pp. 3239- 3244.

[5] S. L. Wang and A. Jafari, "Hiding Sensitive Predictive Association Rules", IEEE International Conference on Systems, Man and Cybernetics, Vol. 1, Oct 2005, pp. 164- 169.

[6] V. Verkios, E. Bertino, I. G. Fovino, L. P. Provenza, Y. Saygın and Y. Theodoris, "State-of-the-art in Privacy Preserving Data Mining", SIGMOD Record, Vol. 33, No. 1, March 2004, pp. 50-57

[7] R. Srikant and R. Agrawal, "Mining Quantitative Association Rules in Large Relational Tables", Proceedings of ACM SIGMOD, 1996, pp. 1-12.

[8] L.A. Zadeh, "Fuzzy Sets," Information and Control, Vol. 8, 1965, pp. 338-353.

[9] T. P. Hong, C. S. Kuo, S. C. Chi, "Mining association rules from quantitative data", Intell. Data Anal 3 (5), 1999, pp. 363–376.

[10] M. Kaya, R. Alhajj, F. Polat and A. Arslan, "Efficient Automated Mining of Fuzzy Association Rules," Proc. of DEXA, 2002.

[11] Chen G., Yan P., Kerre E.E,"*Computationally Efficient Mining for Fuzzy Implication-Based Association Rules in Quantitative Databases*", International Journal of General Systems, Vol. 33, No. 2-3, 2004. pp. 163-182.

[12] T. Berberoglu and M. Kaya, "Hiding Fuzzy Association Rules in Quantitative Data"**,**The 3rd International Conference on Grid and Pervasive Computing Workshops, May 2008, pp. 387-392.

[13] J. Han and M. Kamber, "Data Mining: Concepts and Techniques", 2/e, Morgan Kaufmann Publishers, March 2006, pp. 230.

[14] http://mlearn.ics.uci.edu/databases/breast-cancerwisconsin/breast-cancer-wisconsin.da