Int. J. Advance Soft Compu. Appl, Vol. 7, No. 1, March 2015
ISSN 2074-8523

An Algorithm for Hiding Sensitive Frequent Itemsets

Maryam Nourafkan, Hamid Rastegari, and Mohammad Naderi Dehkordi

Department of Computer Engineering,
Islamic Azad University–Najafabad Branch, Isfahan, Iran
e-mail: Norafkan_mary@yahoo.com, Rastegari@iaun.ac.ir

Abstract

Association rule mining is an important data-mining technique that finds interesting association among a large set of data items. Since it may disclose patterns and various kinds of sensitive knowledge that are difficult to find otherwise, it may pose a threat to the privacy of discovered confidential information. This study investigates how to shelter certain information and/or confidential knowledge in the data set and how to create a new database for non-confidential access. The proposed approach uses the data distortion technique. In this connection, sensitive representative rules are mined based an algorithm named GSRR. Then, in immunization phase, an algorithm named EDSR is presented. In this algorithm, the procedure of hiding the sensitive itemsets is carried out through the reduction of sensitive representative rules confidence rate. Regarding this, the changes occur on the right hand side items of the rules. These changes occur on transactions which fully support sensitive representative rules, and among the transactions, a transaction is selected for the change which has the fewest number of items. The goal is having the minimum change on database. Performance comparison of the recommended algorithm and the two benchmark algorithms on the dense database of Chess, illustrated that the proposed algorithm run time has considerably decreased in comparison with the benchmark algorithms. Also, regarding the number of lost rules, the algorithm is more practical than the benchmark algorithms.

Keywords: *Hiding Sensitive Itemsets, Representative Rules, Privacy Preserving Data Mining.*

1 Introduction

The data explosion in recent years urges the need of data mining in the form of tools for the better management of immense data and the investigation of the new relation and information. In cases, organizations require to reveal their data. Since confidential data might exist here in this giant size of data, organizations might not be willing to expose the confidential sections. This study investigates how to shelter certain information and/or confidential knowledge in the organization data set and how to create a new database for non-confidential access. This investigation encompasses sensitive association rules gained through data mining algorithm operations. This method of concealing data is founded on distortion-based technique. At first, the data mining owner identifies the sensitive item sets. Then, sensitive representative rules which includes the members of this set on the right hand side is hidden through the recommended algorithm. The rest of the paper is organized as follows. Section 2 presents the statement of the problem and the notation used in the paper. Section 3 presents a review of related performed work. Proposed algorithms are presented in section 4. Section 5 shows example of the proposed algorithms. Section 6 compares and evaluates the proposed algorithms with algorithms 1.b [3,15] and DSR [16,17]. And section 7 presents future works, discussions and conclusion.

2 Problem Statement

Let $I = \{i_1, i_2 \dots, i_m\}$ be a set of m distinct literals, called items. Given a set of transactions D, where each transaction T is a set of items such that $T \subseteq I$. An association rule is an implication of the form $X \rightarrow Y$ where $X \subset I, Y \subset I, X \cap Y = \phi$. X and Y are called antecedent/body and consequent /head of the rule respectively. Strength of a rule whether it is strong or not is measured by two parameters called support and confidence of the rule. These two parameters help in deciding the interestingness of a rule [2, 7, 15]. For a given rule $X \rightarrow Y$ support is the percentage of transaction that contains both X and Y $(X \cup Y)$ or is the proportion of transactions jointly covered by the LHS and RHS and is calculated as:

$$Supp(X \rightarrow Y) = Supp(X \cup Y) = \frac{|X \cup Y|}{|N|} \tag{1}$$

Where, N is the number of transactions. Confidence is the percentage for a transaction that contains X also contains Y or is the proportion of transactions covered by the LHS that are also covered by the RHS and is calculated as

$$Conf(X \rightarrow Y) = \frac{|X \cup Y|}{|X|} = \frac{Supp(X \cup Y)}{Supp(X)} \tag{2}$$

For a database of transactions with certain sets of items, there can be too much association rules potentially. A rule is significant if its support and confidence is higher than the user specified minimum support threshold (MST) and minimum confidence threshold (MCT). In this way, algorithms do not retrieve all the association rules that may be derivable from a database, but only a very small subset that satisfies the minimum support and minimum confidence requirements set by the users. An association rule-mining algorithm works as follows. It finds all the sets of items that appear frequently enough to be considered relevant and then it derives from them the association rules that are strong enough to be considered interesting. We aim at preventing some of these rules that we refer to as "sensitive rules", from being disclosed. The problem can be stated as follows: Given a database D, a set R of relevant rules that are mined from D and a subset RH of R, how can we transform D into D' a database in such a way that the rules in R can still be mined, except for the rules in RH?

There are two main approaches that can be adopted when we try to hide a set RH of rules (prevent them from being discovered by association rule mining algorithms): (a) we can either prevent the rules in RH from being generated, by hiding the frequent sets from which they are derived, or (b) we can reduce the confidence of the sensitive rules, by bringing it below a user-specified threshold (MCT) [2, 7, 15].

3 Background and Related Work

There are mainly 3 approaches for association rule hiding (i) Exact approach (ii) Border based approach (iii) Heuristic approach. In following, overview of these approaches is given in brief [7, 14].

Exact approach: This approach contains none heuristic algorithms which formulates the hiding process as a constraints satisfaction problem or an optimization problem which is solved by integer programming. These algorithms can provide optimal hiding solution with ideally no side effects.

Border based approach: This approach hides sensitive association rule by modifying the borders in the lattice of the frequent and the infrequent itemsets of the original database.

Heuristic approach: This approach involves efficient, fast and scalable algorithms that selectively sanitize a set of transactions from the original database to hide the sensitive association rules. Various heuristic algorithms are based on mainly two techniques: Data distortion technique and blocking technique. Blocking is the replacement of an existing value with a "?". It inserts unknown values in the data to fuzzify the rules. In some applications where publishing wrong data is not acceptable, then unknown values may be inserted to blur the rules. Data distortion is done by the alteration of an attribute value by a new value. It changes 1"s to 0"s or vice versa in selected transactions to increase or decrease support or confidence of sensitive rule. Heuristic algorithms cannot give an optimal solution because of undesirable side effects to none sensitive rules, e.g. lost rules and new rule. Algorithms proposed using heuristic approach can be divided into rule hiding and itemset hiding algorithms. Atallah et al [1] for the first time provided an algorithm for hiding association rule by reducing the amount of support. Reduction of the amount of support in the set of Items is done using a data base Lattice graph.

Dasseni et al [3] aimed at reducing the effect on non-sensitive Items. Decreasing the amount of Confidence with increasing the amount of supporting Antecedent (LHS) rules, through transactions which Partially Support adhere by the rules until the time Confidence reaches below its minimum Confidence and hides the rules. Verykios et al [15] aimed at hiding an Item that has the maximum support among short length transactions. Hiding set of items in a round robin method. Items are eliminated through the round robin method until the amount of support reaches below the MCT. Olivera & Zaiane [9, 10] for the first time proposed the method of Multiple Rule Hiding. To hide there is a need to scan the dataset twice, regardless of the number of Sensitive Items. The first scan is to make index files to speed up the process of finding sensitive transactions and allow for efficient retrieval of data. The second scan is to apply algorithm Dataset selectively. Shah, Takkar and Ganatra [14] proposed two association rule hiding algorithms, ADSRRC (Advanced Decrease Support of R.H.S. items of Rule Cluster) and RRLR (Remove and Reinsert L.H.S. of Rule), based on heuristic approach. Both algorithms are based on algorithm DSRRC (Decrease Support of R.H.S. items of Rule Cluster) proposed in [8]. Algorithm DSRRC depends on ordering of transactions for removing items from database. Also it requires sorting of database each time item is removed from database. Algorithm ADSRRC is proposed to overcome these limitations. Algorithm DSRRC cannot hide rule having multiple R.H.S. items. To overcome this limitation algorithm RRLR is proposed. Jain et al [4] proposed approach uses the data distortion technique where the position of the sensitive items is altered but its support is never changed. The size of the database remains the same. The proposed heuristics use the idea of representative rules to prune the rules first and then hides the sensitive rules. Advantages of the proposed approach is that the support of the sensitive item(s) is neither increased nor decreased as done in existing approaches and the size of the database is kept same. Support of the sensitive item(s) is kept same and simply its position have been changed i.e. it is being deleted from one transaction and added to some other transaction in which it does not exist. Another advantage of this approach is that it hides maximum number of rules in minimum number of alterations in the database. Wang and Jafari [16, 17] assumed that only sensitive items are given and propose two algorithms, ISL (Increase Support of LHS) and DSR (Decrease Support of RHS), to modify data in database so that sensitive predicative rules containing specified items on the left hand side of rule cannot be inferred through association rule mining.

Discovering association rules between items in a large database is an important database mining problem. The number of association rules may be huge. Kryszkiewicz [5] introduced a notion of a cover operator which transforms an association rule into the set of association rules by syntactic transformation of the initial rule. Representative association rules are defined as a least set of rules that covers all association rules satisfying certain user specified support and confidence. A user may be provided with a set of representative association rules instead of the whole set of association rules. In this paper, an algorithm for computing representative association rules is offered. In this paper, to check whether a candidate rule is representative the algorithm required comparing the rule with longer representative rules, which was quite time consuming operation. Kryszkiewicz [6] investigated some properties of representative association rules and propose a new efficient algorithm for representative association rules mining. The new algorithm generates representative rules independently from other representative rules. Unlike the algorithm proposed in [5], the new algorithm exploits solely the information about the supports of frequent itemsets.

4 Proposed Algorithm

In this paper, an algorithm for hiding sensitive association rules based on data distortion technique is presented. In this connection, representative rules are mined based an algorithm named Generate Sensitive Representative Rules (GSRR). Then, in immunization phase, an algorithm named Extended Decrease Support of R.H.S (EDSR) is presented. In this algorithm, the procedure of hiding the sensitive itemsets is carried out through the reduction of sensitive representative rules confidence rate. In this way, the changes occur on the right hand side items of the rules. These changes occur on transactions which fully support sensitive representative rules, and among the transactions, a transaction is selected for the change which has the fewest number of items. The goal is having the minimum change on data base. Performance comparison of the recommended algorithm and the two benchmark algorithms on the dense database of Chess, illustrated that the recommended algorithm run time has considerably decreased in comparison with the benchmark algorithms. Also, regarding the number of lost rules, the recommended algorithm is more practical than the benchmark algorithms.

4.1 Cover operator

A notion of a cover operator was introduced in [5, 6, 13] for deriving a set of association rules from a given association rule without accessing a database. The cover C of the rule $r: X \to Y, Y \neq \emptyset$ is defined as follows:

$$C(r: X \to Y) = \{X \cup U \to V \mid U, V \subseteq Y, U \cap V = \emptyset, and\ V \neq \emptyset\} \quad (3)$$

Each rule in $C(r: X \to Y)$ consists of a subset of items occurring in the rule $X \to Y$. The antecedent of any rule r covered by $X \to Y$ contains X and perhaps some items from Y, whereas r's consequent is a non-empty subset of the remaining items in Y. It was proved in [5] that each rule r in the cover $C(r')$, where r' is an association rule having support s and confidence c, belongs in $AR(s,c)$. Hence, if r belongs in $AR(s,c)$ then every rule r' in $C(r)$ also belongs in $AR(s,c)$.

4.2 Representative association rules

In this section we describe a notion of representative association rules which was introduced in [5, 6, 13]. Informally speaking, a set of all representative association rules is a least set of rules that covers all association rules by means of the cover operator. A set of representative association rules with minimum support s and minimum confidence c will be denoted by $RR(s,c)$ and defined as follows:

$$RR(s,c) = \{r \in AR(s,c) \mid \nexists r' \in AR(s,c), r \neq r' \text{ and } r \in C(r')\} \tag{4}$$

If s and c are understood than $RR(s,c)$ will be denoted by RR. Each rule in RR is called a representative association rule. By the definition of RR no representative association rule may belong in the cover of another association rule.

4.3 GSRR algorithm

GSRR algorithm which is a modification of the FastGenAllRepresentatives given below: [6]

```
Algorithm GSRR
INPUT: (1) all frequent itemsets $F$,
           (2) a min_confidence $c$,
           (3) a set of items $X$;
OUTPUT: sensitive representative rules SRR;
1. forall $x \in X$ do begin
2. If ($x$ isn't in $F_1$) then $X = X - \{x\}$;
3. endfor;
4. If ($X$ is empty) then EXIT;
5. Select all itemsets from $F$ which have $x$ and store in $FS$;
6. forall $Z \in FS$ do begin
7. $K = |Z|$; $maxsupp = max(\{supp(Z') \mid Z \subset Z' \in F_{K+1}\} \cup \{0\})$;
8. If $Z.supp \neq maxsupp$ then begin
9. $A_1 = \{\{Z[1]\}, \{Z[2]\}, \ldots, \{Z[k]\}\}$;
/* loop1 */
10. for ($i = 1$; ($A_i \neq \emptyset$) and ($i < K$); $i++$) do begin
11. forall $X \in A_i$ do begin
12. find $Y \in F_i$ such that $Y = X$;
```

```
$XCount = Y.Count;$
$if\ (Z.supp\ /\ XCount \geq c)\ then\ begin$
$if\ (maxsupp/XCount < c)\ then$
$print\ (X, "\rightarrow", Z\backslash X, "with\ support:", Z.supp, "and\ confidence:", Z.supp\ /\ XCount);$
$A_i = A_i \setminus \{X\};$
$endif;$
$endfor;$
$A_{i+1} = AprioriGen(A_i);$
$endfor;$
$endif;$
$endfor;$
```

A brief description of important steps of the algorithm is given below:

The GSRR algorithm computes sensitive representative association rules from each itemset in FS. Let Z be a considered itemset in FS. Only k-rules, $K = |Z|$ are generated from Z. First, $maxsupp$ is determined as a maximum from the supports of these itemsets in F_{k+1} which are supersets of Z. If there is no superset of Z in F_{k+1} then $maxsupp = 0$. Loop1 starts. In general, the i-th iteration of Loop1 looks as follows: Each candidate $X \rightarrow Z\backslash Y$, where $X \subset Z$ belongs in i-itemsets A_i, is considered. Z is frequent, so X, which is a subset of Z, is also frequent. In order to check if $X \rightarrow Z\backslash Y$ is an association rule its confidence: $supp(Z)\ /\ supp(x)$ has to be determined. $supp(Z) = Z.supp$, while $supp(x)$ is computed as Y. Only association rules that satisfy $maxsupp \leq s$ or $maxsupp/XCount < c$ are representative. The antecedent X of each association rule $X \rightarrow Z\backslash Y$ is removed from A_i. Having found all representative k-rules with i-antecedents from Z, (i+1)-itemset antecedents A_{i+1} are built from A_i by the AprioriGen function given in [5, 6].

4.4 EDSR algorithm

In immunization phase, the EDSR algorithm hides sensitive representative association rules which have sensitive items on the right hand side. The algorithm is given below:

```
$Algorithm\ EDSR$
$INPUT: (1)\ sensitive\ representatives\ rules\ SRR,$
        $(2)\ a\ min\_confidence\ c,$
```

```
          (3) a set of items $X$;
OUTPUT: the database $D'$,
            where rules containing $X$ on right hand side of the rule will be hidden;
1. forall $x \in X$ do begin
    2. Select all r from SRR which have x on the right hand side and store in U;
    3. forall $r \in U$ do begin
        4. Compute Confidence of rule
        5. If (confidence of rule $<$ c) then
            6. Choose next r in U;
        7. Else begin
            8. Find $TR = \{t$ in $D/$ t fully support $r\}$;/
                        /transactions which have all items of rule
            9. If (TR is empty) then go to next r in U;
            10. Sort TR in ascending order by the number of items;
            11. While(confidence of rule $\geq$ c or TR is not empty) do begin
                12. Choose first t from TR;
                13. Modify t so that x is not supported; //delete x from r
                14. Compute confidence of rule;
                15. Remove t from TR and save first t;
            16. endwhile;
        17. endelse;
    18. endfor;
    19. Remove x from X;
20. endfor;
```

A brief description of important steps of the algorithm is given below:

The EDSR algorithm hides sensitive representative association rules which have sensitive items on the right hand side. Step 2, selects all rules from the set of SRR's, which have sensitive item x on the right hand side. Step 8-13, deletes the sensitive item(s) from the transaction that completely supports the SRR i.e. it contained all the items in of SRR selected. Step 14, recomputed the confidence of the rule.

5 Example

The proposed algorithms can be illustrated with the following example for a given set of transactional data given in Table 1. Considering $s = 3$, the following itmesets are mined:

Table 1: Database

TID	A	B	C	D	E	F	G	H
T1	1	1	1	1	1	0	0	0
T2	1	1	1	1	1	1	0	0
T3	1	1	1	1	1	0	1	1
T4	1	1	0	0	1	0	0	0
T5	0	1	1	1	1	0	1	1

$$F = \{A = 4, B = 5, C = 4, D = 4, E = 5, AB = 4, AC = 3, AD = 3, AE = 4, BC = 4, BD = 4, BE = 5, CD = 4, CE = 4, DE = 4, ABC = 3, ABD = 3, ABE = 4, ACD = 3, ACE = 3, ADE = 3, BCD = 4, BCE = 4, BDE = 4, CDE = 4, ABCD = 3, ABCE = 3, ABDE = 3, ACDE = 3, BCDE = 4, ABCDE = 3\}$$

Applying the GSRR algorithm on the database represented in Table 1 for minimum confidence $c = 0.75$ and a set of sensitive items $X = \{A\}$, the following sensitive representative rules are found:

$$SRR = \{A \rightarrow BCDE, C \rightarrow ABDE, D \rightarrow ABCE, B \rightarrow AE, E \rightarrow AB\}$$

Applying the EDSR algorithm on the SRR, minimum confidence c and a set of sensitive items X, the rules containing sensitive items in the RHS are:

$$U = \{C \rightarrow ABDE, D \rightarrow ABCE, B \rightarrow AE, E \rightarrow AB\}$$

Start from $C \rightarrow ABDE$, then delete A from a transaction in which $A, B, C, D,$ and E are present. This results in modification of the database by changing the transaction T1 to $BCDE$. Out of 4 rules containing sensitive items in the RHS all of them are hidden. The modified database given in Table 2:

Table 2: Modified database

TID	A	B	C	D	E	F	G	H
T1	0	1	1	1	1	0	0	0
T2	1	1	1	1	1	1	0	0
T3	1	1	1	1	1	0	1	1
T4	1	1	0	0	1	0	0	0
T5	0	1	1	1	1	0	1	1

6 Comparison and Evaluation of the Proposed Algorithm

In this paper, nine experiments to evaluate the proposed approach are presented. We performed our experiments on a system with a seven-core processor and with 1 GB of main memory, under Windows 7 operating system. We used dataset Chess which is available through FIMI [18] and its properties are summarized in Table 3. Table 4 presents the result of mining of it as well.

Table 3: Properties of dataset

Dataset	Number of records	Number of items	Avg. Items.
Chess	3195	75	37

Table 4: Result of mining on dataset

Dataset	MCT	MST	Extracted rules from the original database	Extracted frequent itemsets from the original database
Chess	96%	95%	344	77

The amount of hiding failure for hiding 2, 4, and 6 sensitive items with three algorithms in the Chess dataset is equal to zero, since there is a loop in all three algorithms and the loop would not end unless the rule is covered. (See Fig. 1)

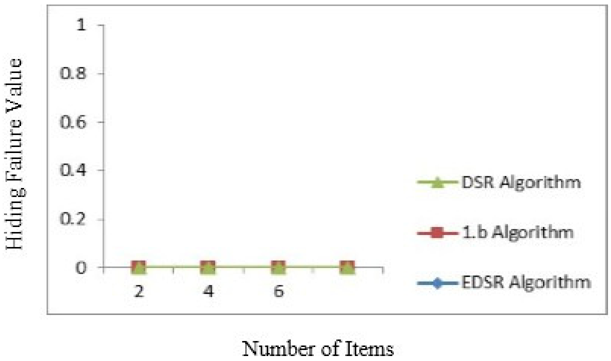

Fig. 1: Failure hiding for hiding 2, 4, and 6 sensitive items in the Chess dataset

The proposed algorithm focuses on a set of sensitive representative association rules which are a least set of rules that covers all association rules by means of the cover operator. It uses the idea of representative rules to prune the rules first and then hides the sensitive rules. Due to this property by hiding a sensitive representative rule, sub rules that are covered by this rule will be concealed as well. This leads to reduce run-time, and changes which are applied to the database. on the contrary, the approaches used by Verykios et al [3, 15] and Wang and Jafari [16, 17] try to hide every single association rule without checking if some rules could be pruned out after some changes have been made in the database while hiding some rules previously. If the number of association rules is too large then the number of passes taken by this approach is equal to the number of rules, which can be a great overhead for hiding algorithms. This process increases the number of lost rules and run-time as well. (See Fig. 2, Fig. 3)

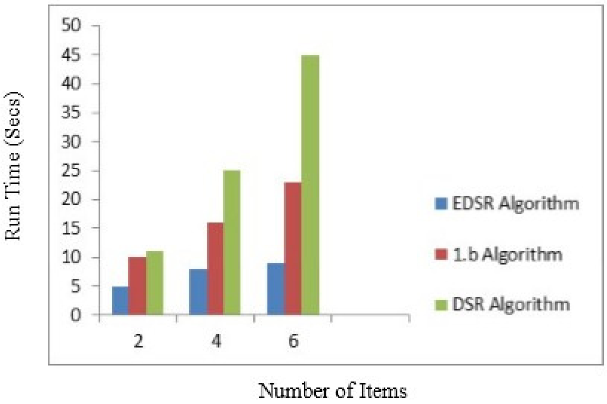

Fig. 2: Runtime for hiding 2, 4, and 6 sensitive items in the Chess dataset

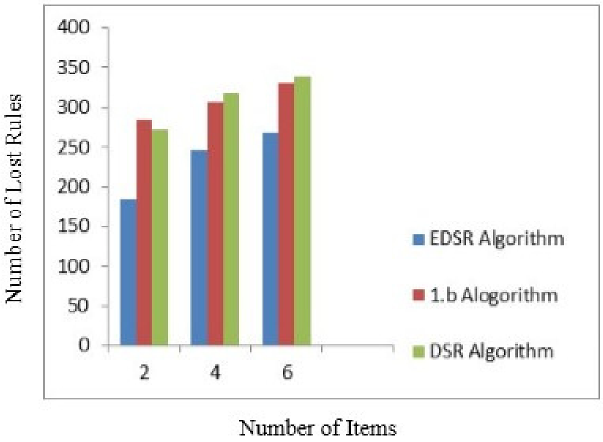

Fig. 3: Rules lost for hiding 2, 4, and 6 sensitive items in the Chess dataset

7 Conclusion and Future Work

In this paper, an algorithm for hiding sensitive association rules based on data distortion technique was presented. In this context, an algorithm named GSRR for pruning extracted rules from a database by using the concept of representative rules was presented. Then, an algorithm called EDSR for hiding sensitive itemset was used. Performance comparison of the proposed algorithm and the two benchmark algorithms on the dense database of Chess, illustrated that run time of the our algorithm has considerably decreased in comparison with the benchmark algorithms. Also, regarding the number of lost rules, the algorithm is more practical than the others.

With regards to these results, we can have a combination of proposed algorithm and ISL algorithm as a future work, to hide all sensitive representative rules not only sensitive representative rules which have sensitive items on the right hand side. In this case, we will benefit from the advantages of both algorithms. Different algorithms to obtain a set of representative association rules are presented .In order to increase the efficiency of the proposed algorithm, we can improve representative association rules mining algorithms.

References

[1] Atallah M, Bertino E, Elmagarmid A, Ibrahim A, Verykios VS. Disclosure limitation of sensitive rules. *Proceedings of the 1999 IEEE Knowledge and Data Engineering Exchange Workshop (KDEX'99)*; 1999. p. 45-52.

[2] Berry MJ, Linoff GS, Data mining techniques: for marketing, sales, and customer relationship management. New Jersey: *The Wiley*; 2004.

[3] Dasseni E, Verykios VS, Elmagarmid AK, Bertino E. Hiding association rules by using confidence and support. *Proceedings of the 4th International Workshop on Information Hiding*; 2001. p. 369-383.

[4] Jain D, Sinhal A, Gupta N, Narwariya P, Saraswat D, Pandey A. Hiding sensitive association rules without altering the support of sensitive item(s). *International Journal of Artificial and Applications (IJAIA)*. 2012; 3(2): 75-84.

[5] Kryszkiewicz M. Representative association rules. *Proceedings of the 1998 Springer-Verlag Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'98)*; 1998. p. 198-209.

[6] Kryszkiewicz M. Fast discovery of representative association rules. *Proceedings of the 1998 Springer-Verlag Rough Sets and Current Trends in Computing (RSCTC'98)*; 1998. p. 214-221.

[7] Luo Y, Zhao Y, Le J. A survey on the privacy preserving algorithm of association rule mining. *Proceedings of the 2009 IEEE 2nd International*

Symposium on Electronic Commerce and Security (ISECS'09); 2009. p. 241-245.

[8] Modi CN, Rao UP, Patel DR. Maintaining privacy and data quality in privacy preserving association rule mining. *Proceedings of the 2010 IEEE International Conference on Computing Communication and Networking Technologies (ICCCNT'10)*; 2010. pp. 1-6.

[9] Oliveira SRM, Zaiane OR. Privacy preserving frequent itemset mining. *Proceedings of the 2002 IEEE International Conference on Privacy, Security and Data Mining (PSDM'02)*; 2002; Maebashi City, Japan. p. 43-54.

[10] Oliveira SRM, Zaiane OR. Protecting sensitive knowledge by data sanitization. *Proceedings of the IEEE International Conference on Data Mining (ICDM'03)*; 2003. p. 211-218.

[11] Pasquier N, Bastide Y, Taouil R, Lakhal L. Efficient mining of association rules using closed item set lattices. *Information Systems*. 1999; 24(1): 25-46.

[12] Pasquier N, Bastide Y, Taouil R, Lakhal L. Discovering frequent closed itemsets for association rules. Proceedings of the 1999 Springer- Heidelberg International Conference on Database Theory (ICDT '99); 1999. p. 398-416.

[13] J. Saquer, J.S. Deogun, Using Closed itemsets for discovering representative association rules, *Proceedings of the 2000 Springer-Verlag International Symposium on Methodologies for Intelligent Systems (ISMIS'10)*, p. 495-504, 2010.

[14] K. Shah, A. Thakkar and A. Ganatra, Association Rule Hiding by Heuristic Approach to Reduce Side Effects & Hide Multiple R.H.S. Items. *International Journal of Computer Applications*. 2012; 45(1): 1-7.

[15] Verykios VS, Emagarmid AK, Bertino E, Saygin Y, Dasseni E. Association rule hiding. *IEEE Transactions on Knowledge and Data Engineering*. 2004; 16(4): 434-447.

[16] Wang SL, Jafari A. Hiding sensitive predictive association rules. *IEEE International Conference on Systems, Man and Cybernetics*, p. 164-169, 2005.

[17] Wang SL, Parikh B, Jafari A. Hiding informative association rule sets. *Journal of Expert Systems with Applications*. 2007; 33(2): 316-323.

[18] Available: http://fimi.cs.helsinki.fi/data/