Volume 3, Issue 10, October 2013 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Privacy Preserving Generalized Coherent Rule Mining

Sharada Narra*
Department of Information Technology
ANITS, Vizag, A.P., India

Madhavi Dabbiru
Department of Computer Science Engineering
Dr.L.B.College of Engineering, Vizag, A.P, India

Abstract— In coherent rule mining, an association rule should only be reported when there is enough interest gain claimed during inference analysis in the data. Coherent rule mining algorithm contemplated by Sim et al [11] discovers interesting patterns based on the logical evidences in the data irrespective of how frequent they are. The discovery of coherent rules does not require user to preset a minimum support threshold. The authors proposed a generalized coherent rule mining algorithm which discovers all possible interesting patterns for the given transactional dataset. The pruning process is introduced to speed up the generation of coherent patterns by removing candidate itemsets that may not become coherent itemsets. The authors also developed a novel privacy preserving algorithm for coherent rule mining framework. Coherent rule mining discovers both frequent and infrequent interesting patterns. Hence it is a new task to hide sensitive coherent itemsets. The traditional privacy preserving algorithms are not applicable for coherent itemsets. The objective of the proposed framework is to obtain a balance between privacy preserving and discovery of information.

Keywords— Data Mining, Association Rule Mining, Coherent Rules, Sensitive itemsets , Privacy Preserving data mining

I. INTRODUCTION

The problem of mining association rules over customer transactions was introduced by Agrawal et al in [1]. Apriori is a classical algorithm for frequent item set mining and learning association rules that satisfy the predefined minimum support and confidence from a given database. However discovery of associations based on preset threshold requires in-depth domain knowledge. One of the main issues of the existing pattern mining techniques is how to define appropriate minimum support and confidence measures. Also some items are rare in nature which is found with less support count in a dataset. These items are called rare items [4]. If a single minimum support threshold is used and set high those association rules consisting of rare items will not be discovered. Coherent rule mining algorithm contemplated by Sim et al [11] discovers interesting patterns based on the logical evidences in the data irrespective of how frequent they are. This concept does not require the user to specify the minimum support threshold. This approach considers that an association rule should only be reported when there is enough interest gain claimed during inference analysis in the Coherent rule mining avoids reporting the rule $P \rightarrow Q$ if there is a stronger association between item $P \rightarrow \neg Q$, thereby eliminating redundant rules. These association rules detect interesting item sets that are frequently and infrequently observed in a set of transaction records.

Contemporary advances in data mining and machine learning algorithms have introduced new problems in privacy protection. To do so, original database is converted into a sanitized one by inserting minimum number of dummy transactions. On the other hand, such an approach must hold the following restrictions: (1) the impact on the non-restricted data has to be minimal and (2) an appropriate balance between a need for privacy and knowledge discovery must be guaranteed. In this paper, focus is on privacy preserving mining. There are many Privacy preserving data mining algorithms in the literature. These algorithms sanitize the database by padding some additional transactions wherein frequent itemsets are converted into infrequent itemsets, for a given minimum support threshold. The sanitized database thereby hides sensitive itemsets. Coherent rule mining discovers both frequent and infrequent itemsets without user specified minimum support threshold. Hence it is a challenge to hide sensitive itemsets using the traditional privacy preserving algorithms. In this paper the authors approach is two phases: In the first phase, authors propose generalized coherent rule mining algorithm which discovers all possible interesting patterns for the given transactional dataset. The pruning process is introduced to speed up the generation of coherent patterns by removing candidate itemsets that may not become coherent itemsets.In the second phase, the authors developed a new privacy preserving algorithm for coherent rule mining.The organization of the rest of the paper is as follows. In Section 2, we give a formal statement of the problem and the notation used in the paper. In Section 3, we present our generalized algorithm for mining coherent rules. Section 4 presents the proposed algorithm for hiding sensitive coherent itemsets. Section 5 discusses with an example of the proposed algorithm. Concluding remarks and future work are described in section 6 followed by references.

II. RELATED WORK

Many data mining methods have been proposed for deriving association rules from transaction databases. The initial algorithm proposed by Agrawal et al. in [1] is based on the concept of large itemsets to find association rules in transaction data. In [3] K-optimal rule discovery algorithm finds the k rules that optimize a user-specified measure of rule value with respect to user specified constraints.Use of single and lower minimum support threshold, would result in too many uninteresting association rules. Liu et al., proposed multiple minimum thresholds algorithm which sets a minimum support for each item in a data set [4]. MIS results in association rules with items which occurs infrequently and below a minimum support threshold. Brin et al., showed that association rules discovered using a support and confidence framework may not be correlated in statistics [5].The coherent rule mining algorithm proposed by Sim et al. is based on the properties of preposition logic. [11]This discovers the relationships among items in the transactional database without the necessity of user specified minimum support threshold.

One simple way to hide some sensitive patterns is to decrease their support in a given database. This procedure of altering the transactions is called the sanitization process and was introduced in [14]. To do so, a small number of transactions have to be modified either by deleting one or more items from them or even changing items in transactions.

III. GENERALIZED COHERENT PATTERN MINING

This paper proposes an Apriori like approach namely Generalized Coherent Rule Mining Algorithm based on preposition logic for mining patterns and protecting sensitive itemsets in transactional databases. Since checking all possible candidate itemsets for interestingness is time consuming, the proposed approach validates candidate itemsets against a qualification criterion before checking for logical implications. This criterion is used for removing itemsets that cannot become highly coherent itemsets which reduces the search space and speeds up mining process. Then set cardinality of qualified itemsets are calculated and verified whether they satisfy the conditions of logical equivalence. By utilizing the coherent rule concept the present study proposes generalized coherent rule mining algorithm that uses four conditions in the mining process and discovers coherent rules from transactional database.

A. Terminology

The coherent rule mining proposed by Sim et al. is based on the prepositional logic. This approach maps the association rules to equivalences. In transactional database a item set has two states, an item set can be present or absent. It is analogous to a proposition that can be either true or false.

X and Y are two itemsets , then association rule $X \rightarrow Y$ is mapped to the implication $p \equiv q$ if both itemsets are present in single transactional record.In multiple transactions association rules are mapped to implications as follows:

$X \rightarrow Y$ is mapped to an implication $p \rightarrow q$ if and only if

- $S(X,Y) > S(X, \neg Y)$
- $S(X,Y) > S(\neg X,Y)$
- $S(X,Y)>S(\neg X, \neg Y)$

Equivalences	$p \equiv q$	$\neg p \equiv \neg q$
Association Rules	$X \rightarrow Y$	$\neg X \rightarrow \neg Y$

Constraint based mapping of associations to Equivalences is shown in Table 01. When association rules are mapped to implications based on comparison between supports then it is pseudo implication. Sim et al. defined a concept called coherent rule based on pseudo implications of equivalences. Two pseudo implications of equivalences exist as a pair because they are created based on the same conditions, i.e if $X \rightarrow Y$ coherent rule exists, then $\neg X \rightarrow \neg Y$ also exists.

True or False on association rules	**Required conditions (to map associations to equivalences)**	
T	$X \rightarrow Y$	$\neg X \rightarrow \neg Y$
F	$X \rightarrow \neg Y$	$\neg X \rightarrow Y$
F	$\neg X \rightarrow Y$	$X \rightarrow \neg Y$
T	$\neg X \rightarrow \neg Y$	$X \rightarrow Y$

TABLE 1

The following conditions must be met for coherent rule:

- $S(X,Y) > S(X, \neg Y)$, $S(X,Y) > S(\neg X,Y)$, $S(\neg X, \neg Y) > S(X, \neg Y)$, $S(\neg X, \neg Y) > S(\neg X,Y)$

By utilizing the coherent rule concept the present study proposes generalized coherent rule mining algorithm.

B. Mapping using Set Cardinality Representation.

Let X and Y are itemsets. The cardinality of the union of two itemsets X and Y is given as $|X \cup Y| = |X| + |Y| - |X \cap Y|$ Where $|X|$ is set of transactions that consists of itemset X , $|Y|$ is set of transactions that consist of itemset Y and $|X \cap Y|$ is set of transactions that consists of both X and Y itemsets.

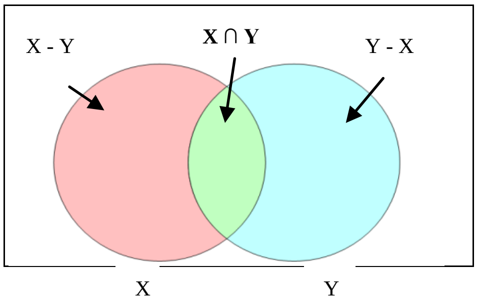

Fig 1. Set representation for coherent rule conditions

The four conditions that must be met for coherent rule are:

- S(X,Y) > S(X, ¬Y), S(X,Y) > S(¬X,Y), S(¬X, ¬Y) > S(X, ¬Y), S(¬X, ¬Y) > S(¬X,Y)

These four conditions can be represented in set cardinality notation as follows:
S(X, Y) = |X∩Y|, the number of elements in the set of transactions that consists of both itemsets X and Y.
S(X, ¬Y) = |X-Y|, the number of elements in the set of transactions that consists of itemset X but not itemset Y.
S (¬X, Y) = |Y-X|, the number of elements in the set of transactions that consists of itemset Y but not X.

S (¬ X, ¬Y) = μ - |X∪Y|, the number of elements in the neither set of transactions that consists of itemsets X nor Y.

The four coherent rule conditions in set cardinality notation.

- |X∩Y| > |X-Y|, |X∩Y| > |Y-X|, μ - |X∪Y| > |X-Y|, μ - |X∪Y|) > |Y-X|

By utilizing these four conditions proposed algorithm uses iterative approach for discovering the coherent itemsets. The candidate k-itemsets are generated from highly coherent (k-1)-itemsets. Each candidate k-itemsets is tested against the four set cardinality conditions which is time consuming; hence a prequalification condition is used to reduce the search space and time complexity. Given candidate itemset S, all possible candidate itemsets {X, Y} can be formed, where X and Y are the subsets of S, and (X∩Y) =∅.The support of an itemset {X, Y} never exceeds the support of its subsets according to anti-monotone property of support. $\forall X, Y : (X \subseteq Y) \Rightarrow s(X) \geq s(Y)$
Hence S{X,Y} ≤ S{X} and S{X,Y} ≤ S{Y} which implies Maximum S{X,Y} = Minimum[S{X}, S{Y}]
The maximum number of elements in the set of transactions that consists of both itemsets X and Y is the minimum number of elements in the set of transactions with itemsets X or itemsets Y.
Maximum |X∩Y| = Minimum (|X|, |Y|)
Hence if candidate itemset satisfies condition Minimum (|X|, |Y|) > |X-Y| then the itemset is tested for set cardinlity conditions. If a candidate itemset does not satisfy this condition, then this itemset cannot generate a coherent itemset, and is removed without testing for the coherent condition thereby enhancing the efficiency in the generation of coherent itemsets.

C . Proposed Algorithm
Algorithm1 : Generalized search for coherent patterns
Input: Transactional database D
Output: Coherent Itemset CI
C_k: Candidate itemset of size k //C_k is generated by joining $L_{k-1}$with itself
CI_k : Coherent itemset of size k
CI ← ∅
Step 1: CI_1 = {items with support count};
For (k = 1; CI_k !=∅; k++) do
Step 2: { C_{k+1} = candidate coherent itemsets generated from CI_k;
For each candidate coherent itemset (X,Y) in C_{k+1} do
Step 3: If (Minimum (|X|,|Y|) > |X-Y|) then

Step 4: { (i) |X∩Y| > |X-Y|
(ii) |X∩Y| > |Y-X|
(iii) μ - |X∪Y| > |X-Y|
(iv) μ - |X∪Y|) > |Y-X| }
Step 5: {CI_{k+1} = CI_k ∪ (X,Y)}
Step 6: **}** returns $\cup_k CI_k$;

Fig 2. The Generalized Coherent Mining Algorithm

The Generalized Coherent Mining Algorithm in Figure 2 uses iterative approach to generate coherent itemsets. First, the set of candidate 1-itemsets is found by scanning the database to accumulate the count for each item. For each candidate k-itemset S, form all its candidate coherent itemsets{X,Y},where X and Y are subsets of S, and (X∩Y) = ∅. To improve the efficiency in the generation of coherent itemsets, each candidate itemset is tested with the prequalification condition. The candidate itemset that do not satisfy the condition is removed, from further consideration which reduces search space. Qualified candidate itemset is tested against, the four set cardinality conditions. If the conditions, are met then candidate coherent k-itemset is added into highly coherent k-itemset CI_k. Candidate k-itemsets are generated from highly coherent (k-1) – itemsets. The process is repeated, until no more coherent k-temsets are found. The efficiency in the discovery of coherent rules is increased greatly by reducing the candidate itemsets to be tested, using anti-monotone property in the pretesting phase. The generalized coherent mining algorithm mines all the k-pattern itemsets from the transactional dataset without user specified constraints.

IV. PRIVACY PRESERVING COHERENT RULE MINING

The privacy-preserving data mining has become an important issue in recent years. In this paper, a novel approach for hiding sensitive itemsets in coherent rule mining framework is proposed. It computes the minimum number of transactions to be inserted into the original database for totally hiding sensitive itemsets. With the newly inserted transactions, set cardinality conditions for the sensitive itemsets are not satisfied, thereby hides them.

Algorithm 2: Sensitive itemset hiding for coherent rule mining
Input: Transactional database D, Sensitive Itemset S
Output: Sanitized database D'
Begin
// Selecting the candidate itemsets for each sensitive itemset
Step1: For i = 1 to n sensitive coherent k-itemset S , j = 1 to 2,
{ form all its candidate coherent itemsets{X,Y},where X and Y are subsets of S, and (X∩Y) = ∅,which satisfies coherent set cardinality conditions.
Step2: Calculate SCD = |X∩Y| - |X-Y|, put the candidate itemsets {X ¬Y} with SCD into the set of Insert_Items[i,j]
Calculate SCD = |X∩Y| - |Y-X|, put the candidate itemsets {¬X, Y} with SCD into the set of Insert_Items[i,j+1]
Arrange the candidate itemsets in the increasing order of their SCD in Insert_Items[i,j] }
// clustering the possible candidate itemsets for all the sensitive itemsets
Step3: Insert_Transactions(Insert_Items[1,1],Original database)
Step 4: For i = 2 to n-sensitive coherent k-itemset S and j=1
{If (Inset_Items[i,j] ∪ Inset_Items[i-1,j]) = true
Append_Itemsets(Insert_Items[i,j],Original database)
Step5 : ElseIf (Inset_Items[i,j+1] ∪ Inset_Items[i-1,j]) = true
Append_Itemsets(Insert_Items[i,j+1],Original database)
Else Insert_Transactions(Insert_Items[i,j],Original database) **}**
End

Figure 3: Privacy preserving coherent rule mining algorithm.

In coherent rule mining framework, minimum number of transactions are inserted into the original database, such that the sanitized database violate any of the four set cardinality conditions, for the user specified sensitive itemset thereby sensitive itemsets will not be reported by the coherent rule mining algorithm.
This algorithm in figure 3 consists of two main steps:
1. Possible candidate itemsets with computed SCD, Support Count Difference for each given coherent sensitive itemset are inserted into Insert_Items[] in the increasing order of SCD. Inserting SCD number of transactions with candidate itemsets {X ¬Y} into the transactional database D disobeys the set cardinality condition |X∩Y| > |X-Y|, thereby hides the sensitive itemset {X, Y}.
2. The candidate itemsets with minimum SCD, generated for each sensitive itemset are clustered to minimize the number of transactions to be inserted into the transactional database D. The candidate itemsets are grouped such that the inserted transaction does not consist of items as well as their negations.
The proposed sensitive itemset mining algorithm hides both frequent and infrequent itemsets effectively by inserting additional transactions into the original database with the Insert_Transactions() function. Append_Itemsets() function groups the candidate itemsets generated for each sensitive itemset into the inserted transactions.

V. DISCUSSION

In this section, an example is given to demonstrate the proposed algorithm. This is a simple example to show how the proposed algorithm can be used to generate coherent itemsets efficiently from a set of transactions. A sample database of 25 transactions with 16 items is used to illustrate proposed algorithms. The dataset is shown in Table 02.
Algorithm 1:
Step 1: The database is scanned to calculate the support count for each item.1-itemsets with support count is shown in Table 03.

Step 2 : Since single item set CI_1 is not null, k=2 is set .The candidate coherent 2-itemsets are generated from coherent 1-itemsets.In this example, a total of 120 candidate coherent 2-itemsets are generated.

Step 3 : Each candidate 2-itemset is tested with pre-qualification condition : (Minimum(|X|,|Y|) > |X-Y|). For example for the candidate itemset {Donuts,Eggs}, the condition is (minimum(10,2) > 8) is false .

TABLE 2. ORIGINAL DATABASE

TID	ITEMS
01	Biscuits, Bread, Cheese, Coffee, Yogurt
02	Bread, Cereal, Cheese, Coffee
03	Cheese, Chocolate, Donuts, Juice, Milk
04	Bread, Cheese, Coffee, Cereal, Juice
05	Bread, Cereal, Chocolate, Donuts, Juice
06	Milk, Tea
07	Biscuits, Bread, Cheese, Coffee, Milk
08	Eggs, Milk, Tea
09	Bread, Cereal, Cheese, Chocolate, Coffee
10	Bread, Cereal, Chocolate, Donuts, Juice
11	Bread, Cheese, Juice
12	Bread, Cheese, Coffee, Donuts, Juice
13	Biscuits, Bread, Cereal
14	Cereal, Cheese, Chocolate, Donuts, Juice
15	Chocolate, Coffee
16	Donuts
17	Donuts, Eggs, Juice
18	Biscuits, Bread, Cheese, Coffee
19	Bread, Cereal, Chocolate, Donuts, Juice
20	Cheese, Chocolate, Donuts, Juice
21	Milk, Tea, Yogurt
22	Bread, Cereal, Cheese, Coffee
23	Chocolate, Donuts, Juice, Newspaper
24	Newspaper, Pastry, Rolls
25	Rolls, Sugar, Tea, Milk

TABLE 3.COHERENT 1-ITEMSETS

Item	Support Count
Biscuits	4
Bread	13
Cereal	10
Cheese	11
Chocolate	9
Coffee	9
Donuts	10
Eggs	2
Juice	11
Milk	6
Newspaper	2
Pastry	1
Rolls	2
Sugar	1
Tea	4
Yogurt	2

TABLE 4. TEST

Set cardinality conditions	**Candidate itemset {Bread,Cereal}**
(i) $\|X \cap Y\| > \|X-Y\|$	9 > 4
(ii) $\|X \cap Y\| > \|Y-X\|$	9 > 1
(iii) $\mu - \|X \cup Y\| > \|X-Y\|$	11 > 4
(iv) $\mu - \|X \cup Y\|) > \|Y-X\|$	11 > 1

DETAILS OF {BREAD,CEREAL}

Hence this candidate itemset is removed and is not tested for coherent set cardinality conditions. In this example a total of 47 candidate coherent itemsets have passed the prequalification test and 73 candidate itemsets are removed.

Step 4: Each prequalified candidate itemset is tested for coherent set cardinality conditions. For example for candidate itemset {Bread,Cereal} the test details are in Table 04.Candidate itemset {Bread,Cereal} satisfy the coherent conditions, hence added to coherent 2-itemset. Similarly all the candidate itemsets which satisfy the coherent conditions are added into coherent 2-itemset as shown in Table 05

TABLE 05 COHERENT 2-ITEMSETS

Coherent 2-Itemsets	**Support Count**
Bread, Cereal	9
Bread, Cheese	8
Bread, Coffee	8
Cheese, Coffee	9
Chocolate, Donuts	7
Chocolate, Juice	7
Donuts, Juice	9
Milk,Tea	4

Step 5: Since 2-itemset CI_2 is not null, k is set to 3. Let $CI_{all} = CI_{all} \cup CI_2$. Step 2 is repeated. Final coherent k-itemsets are listed in Table 06

Algorithm 2: Sensitive itemsets hiding for Coherent Rule Mining

Input : Sensitive Itemsets {(Chocolate, Donuts): 7, (Milk,Tea) :4,(Cheese, Coffee, Bread): 8}, Original Database D.

Output: Sanitized database D'.

Step1: For each sensitive itemset such as {Chocolate,Donuts} form all its candidate coherent itemsets {Chocolate,Donuts},{Donuts,Chocolates} which satisfies coherent set cardinality conditions.

TABLE 06 COHERENT K-ITEMSETS

Coherent 1-itemset		**Coherent 2-itemset**		**Coherent 3-itemset**	
Item	**Count**	**Item**	**Count**	**Item**	**Count**
Biscuits	4	Bread, Cereal	9	Cheese, Coffee, Bread	8
Bread	13	Bread, Cheese	8	Bread, Coffee, Cheese	8
Cereal	10	Bread, Coffee	8	Bread, Cheese, Coffee	8
Cheese	11	Cheese, Coffee	9	Donuts, Juice, Chocolate	7
Chocolate	9	Chocolate, Donuts	7	Juice, Donuts, Chocolate	7
Coffee	9	Chocolate, Juice	7	Chocolate, Donuts, Juice	7
Donuts	10	Donuts, Juice	9		
Eggs	2	Milk, Tea	4		
Juice	11				
Milk	6				
Newspaper	2				
Pastry	1				

Rolls	2				
Sugar	1				
Tea	4				
Yogurt	2				

Step2: To hide sensitive itemset { Chocolate,Donuts } : 7 , Calculate SCD = |Chocolate∩Donuts| - |Chocolate-Donuts| = 5. Put the candidate itemsets {Chocolate, ¬Donuts} with 5 into the set of Insert_Items[1,1]. Sensitive itemset { Chocolate,Donuts } : 7 , can be hidden by inserting 5 transactions with itemset {Chocolate, ¬Donuts}. Similarly Calculate SCD = |Chocolate∩Donuts| - |Donuts-Chocolate| = 4put the candidate itemsets {¬Chocolate, Donuts} with 4 into the set of Insert_Items[1,2]. Sensitive itemset { Chocolate,Donuts } : 7 , can also be hidden by inserting 4 transactions with itemset {¬Chocolate, Donuts } Arrange the generated itemsets in the increasing order of their SCD such as Inserted_Items[1,1] = {¬Chocolate, Donuts} : 4 Inserted_Items[1,2] = {Chocolate, ¬Donuts} : 5
Repeat Steps 1 to 2 for the remaining sensitive itemsets After processing all sensitive itemsets, possible candidate itemsets with their support count differences (SCD) to be inserted into the database D are shown in Table 07

TABLE 07 CANDIDATE ITEMSETS WITH THEIR SCD TO BE INSERTED INTO THE DATABASE D

SI1(Chocolate, Donuts) : 7	**SI2(Milk, Tea) : 4**	**SI3(Cheese, Coffee, Bread) :8**
Insert_Items[1,j]	Insert_Items[2,j]	Insert_Items[3,j]
(¬Chocolate, Donuts) : 4	(Milk, ¬Tea):2	(Bread, ¬Cheese, ¬Coffee) :3
(Chocolate, ¬Donuts) : 5	(¬Milk, Tea) :3	(Cheese, Coffee, ¬Bread) : 7

TABLE 08 INSERTED TRANSACTIONS WITH ITEMSET{DONUTS}

TID	ITEMS
01	Biscuits, Bread, Cheese, Coffee, Yogurt
02	Bread, Cereal, Cheese, Coffee
03	Cheese, Chocolate, Donuts, Juice, Milk
04	Bread, Cheese, Coffee, Cereal, Juice
05	Bread, Cereal, Chocolate, Donuts, Juice
06	Milk, Tea
07	Biscuits, Bread, Cheese, Coffee, Milk
08	Eggs, Milk, Tea
09	Bread, Cereal, Cheese, Chocolate, Coffee
10	Bread, Cereal, Chocolate, Donuts, Juice
11	Bread, Cheese, Juice
12	Bread, Cheese, Coffee, Donuts, Juice
13	Biscuits, Bread, Cereal
14	Cereal, Cheese, Chocolate, Donuts, Juice
15	Chocolate, Coffee
16	Donuts
17	Donuts, Eggs, Juice
18	Biscuits, Bread, Cheese, Coffee
19	Bread, Cereal, Chocolate, Donuts, Juice
20	Cheese, Chocolate, Donuts, Juice
21	Milk, Tea, Yogurt
22	Bread, Cereal, Cheese, Coffee
23	Chocolate, Donuts, Juice, Newspaper
24	Newspaper, Pastry, Rolls
25	Rolls, Sugar, Tea, Milk
26	Donuts, Milk, Bread
27	Donuts, Milk, Bread
28	Donuts, Milk, Bread
29	Donuts

Step3: With Insert_Transactions() funcition insert minimum support count difference(SCD) here SCD = 4, number of transactions with the itemset {¬Chocolate, Donuts} from Insert_Items[1,1] into the transactional database D as shown in Table 08

TABLE 09 FINAL INSERTED TRANSACTIONS WITH ITEMSETS

26	¬Chocolate, Donuts
27	¬Chocolate, Donuts
28	¬Chocolate, Donuts
29	¬Chocolate, Donuts

Step 4: Append_Itemsets() function, append itemsets{ Milk, ¬Tea } into the inserted transactions, since itemsets {¬Chocolate, Donuts} and { Milk, ¬Tea} do not contain negation items, and (Inset_Items[i+1,j] ∪ Inset_Items[i-1,j]) = true .

Steps 3-4 are repeated for every sensitive itemset present in the Insert_items[] set. Then Final Transactions to be inserted into the transactional database D is as shown in Table 09

TABLE 10. FINAL SANITIZED DATABASE

26	¬Chocolate, Donuts, Milk, ¬Tea, Bread, ¬Cheese,
27	¬Chocolate, Donuts, Milk, ¬Tea, Bread, ¬Cheese,
28	¬Chocolate, Donuts, Bread, ¬Cheese, ¬Coffee
29	¬Chocolate, Donuts

Final Sanitized database after inserting Transactions which hides the given sensitive items is shown in Table 10.

VI. CONCLUSIONS

Coherent rule mining generate complete set of interesting rules that are implicational according to the proposition logic. Coherent mining can identify both frequent and infrequent rules, negative association rules. The generation of coherent rules does not require user to specify the minimum support threshold. Two major contributions of this paper are: A generalized coherent rule mining is proposed which discovers all possible interesting k-itemsets that have the properties of propositional logic. To reduce the search space in the generation of coherent rules, pruning phase is introduced, which validities a candidate itemset for a prequalification condition. The itemsets which do not satisfy this condition cannot generate coherent rules and is removed directly without having its set cardinality conditions calculated.

Since calculating set cardinality conditions for each itemset is time consuming, proposed algorithm requires less time to generate coherent rules when compared with Apriori based algorithms.

A novel privacy preserving algorithm is proposed for Coherent Rule Mining. Coherent rule mining discovers both frequent and infrequent itemsets without user specified minimum support threshold. Hence the traditional privacy preserving algorithms are not suitable for hiding coherent rules. The proposed framework aims to obtain a balance between privacy and disclosure of information. The algorithms are tested on synthetic databases and are yet to be experimented with real datasets.

REFERENCES

[1] Agrawal, R., T, Imielinski and A, Swami, 1993,"Mining association rules between sets of items in large databases", Proceedings of the 1993 ACM *SIGMOD International Conference on Management of Data*, May 25-28, ACM, New York, USA., pp: 207-216

[2] Agarwal, R,C., C, Aggarwal and V,V,V, Prasad, 2001,"A tree projection algorithm for generation of frequent item sets",*Journal of parallel and Distributed Computing*, Volume 61, Issue 3,March 2001, Pages 350-371

[3] G.I. Webb and S. Zhang, "K-Optimal Rule Discovery", *Data Mining and Knowledge Discovery*,vol.10,no.1,pp.39-79,2005

[4] B. Liu, W. Hsu, and Y. Ma, "Mining association rules with multiple minimum supports", Proceedings of the *1999 International Conference on Knowledge Discovery and Data Mining,* pp. 337-341, 1999.

[5] S.Brin, R.Motwani, and C.Silverstein "Beyond Market Baskets: Generalizing Association Rules to Correlations", Proc.*1997 ACM SIGMOD Record,* vol.26,pp.255-264,1997

[6] Cai,R., Tung, A.K.H., Zhang,Z., & Hao, Z(2011)."What is unequal among the equals? Ranking equivalent rules from gene expression data", *IEEE Transactions on Knowledge and Data Engineering,* 23(11), 1735-1747

[7] J.Li and Y.Zhang, "Direct Interesting Rule Generation",Proc. *Third IEEE International Conference in Data Mining*, pp. 155-162,2003

[8] Antonie, M.L., Zaiane, O.R.(2004) "Mining positive and negative association rules: An approach for confined rules", In PKDD '04 Proceedings of the *8th European conference on principles and practice of knowledge discovery in databases* (pp.27-38)

[9] Zheng,Z., Zhao,Y., Zuo,z., Cao, L.(2010) "An efficient ga-based algorithm for mining negative sequential patterns", In Proceedings of the *14th Pacific-Asia conference on advances in knowledge discovery and data mining* (pp.262-273)

[10] Marinica , C., & Guillet, F.(2010). "Knowledge-based interactive post mining of association rules using ontologies", *IEEE Transactions on Knowledge and Data engineering*, 22(6),784-797

[11] Alex Tze Hiang Sim, Maria Indrawan, Samar Zutshi, Member, IEEE, and Bala Srinivasan, "Logic Based Pattern Discovery," *IEEE transactions on knowledge and data engineering*, VOL.22, NO. 6, JUNE 2010

[12] Chuang-Kai Chiou, Judy C. R. Tseng "Sorted Compressed Tree: An Improve Method of Frequent Patterns Mining without Support Constraint", *2nd International Conference on Software Engineering and Data Mining (SEDM), 2010*, Issue Date: 23-25 June 2010, On page(s): 328 – 333

[13] Shichao Zhang, Xindong Wu, Chengqi Zhang, Jingli Lu S. "Computing the minimum-support for mining frequent patterns," *Knowledge and Information Systems*, vol. 15, no. 2, 2008, pp. 233-257

[14] Atallah, M., Bertino, E., Elmagarmid, A. K., Ibrahim, M. & Verykios, V. S. (1999), "Disclosure Limitation of Sensitive Rules", *IEEE Knowledge and Data Engineering Workshop*, Chicago, Illinois, USA, pp. 45-52.

