International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

# Hiding Sensitive Association Rules Using EMDSRRC

**Marate Shashank S.[1], Manjusha Yeola[2]**

[1, 2]Department of Computer Engineering, Alard College of Engineering And Management, Pune(M.H),India

**Abstract:** ***Association rules are generated to find out relation between item sets in database. So when it comes to large datasets, generating association rules becomes crucial. There are various techniques which are used to generate association rules such as Apriori algorithm and FP Growth algorithm. To find relation between item sets in database, association rule mining technologies are used. Many organizations uncover their information or data for mutual profits to find some useful information for some decision making purpose and improve their business. But this database may contain some secret data and which the organization doesn't want to uncover. In this paper, a heuristic based algorithm named EMDSRRC (Enhanced Modified Decrease Support of R.H.S. item of Rule Clusters) to hide the sensitive association rules with multiple items in consequent (R.H.S) and antecedent (L.H.S) is proposed. FP growth algorithm is used for generating rules and then selects items based on transactions to hide the sensitive information. We have proposed an algorithm EMDSRRC which uses FP growth algorithm to generate rules to overcome limitation in MDSRRC (Modified Decrease Support of R.H.S. item of Rule Clusters) which uses Apriori algorithm to generate rules.***

**Keywords:** Association Rules, FP growth, Apriori, Sensitivity.

## 1. Introduction

Association rule mining plays crucial role in many organizations. There are organizations where datasets are very large or some with moderate datasets. Association rules are generated from these datasets by analysing datasets. Frequent pattern mining is the widely researched field in data mining because of its importance in many real life applications. Many algorithms are used to mine frequent patterns which give different performance on different datasets. Apriori and FP Growth is the initial basic algorithm used for frequent pattern mining[1]. The premise of this paper is to find major issues/challenges related to algorithms used for frequent pattern mining with respect to transactional database and hiding sensitive association rules. A database layout tells how data is represented. There are two layouts which are in common use, horizontal layout and the vertical layout. In horizontal layout there are two columns. First represents the transaction id and second represents the items bought in that transaction. In vertical layout the first column represent the item id and the second shows the transactions id in which the particular item is bought. There is a third layout also known as projected layout. This is not a physical layout. In this layout the system records only the transaction identifier and associated item. It is a divide and conquer mechanism which reduces the size of database recursively by considering only the longest pattern. A frequent pattern is a pattern which occurs in comparatively more transactions. A frequent itemset is an itemset whose support is greater than some user-specified minimum support[2].

Privacy Preserving Data mining technique is used. In this technique, no private information is opened up[2]. In this paper we have proposed MDSRRC algorithm which removes limitations in DSRRC. Let us take example, a computer store purchases laptops from two companies, X and Y, both can fetch customers database of the store. Now X applies data mining techniques and mines association rules related to Ys products. X had found that most of the customers who buy laptop of the Y also buy accessories. Now X offers home discount on accessories if customer purchases Xs laptop. As result the business of Y goes down. So in this way releasing the database with private information is prone to this problem. This situation promotes research on private rules (or knowledge) hiding in database [5].

In this paper, New introduced EMDSRRC uses FP growth algorithm which uses sensitivity to hide itemsets in rules.

The presented paper is organized in five sections: the first section contains the introduction; the second section presents a brief description of FP Growth algorithm. The third section gives the methodology used for hiding sensitive association rules. The fourth section presents a comparative analysis of the algorithms used under varying conditions. Fifth section gives the conclusion and in the last references is listed.

## 2. Generating Rules Using FP Growth

In existing algorithm MDSRRC Apriori algorithm is used. Apriori is the simplest algorithm which is used for mining of frequent patterns from the transaction database. The main limitation of Apriori algorithm is that the candidate set generation is not time efficient, when there is a large number of patterns and/or long patterns exist. Apriori algorithm uses large item set property which is easy to implement, but it repeatedly scan the database [8]. Apriori takes more time to scan the large Frequents patterns. The frequent pattern tree (FP-tree) is used for storing compressed, important information about frequent patterns, and developed a pattern growth method, FP-growth, for efficient mining of frequent patterns in large databases. FP Growth is basic initial algorithm used for frequent pattern mining. Fp algorithm uses divide and conquer approach and it is more efficient than apriori algorithm and also a time efficient and gives better

performance.

FP tree consists of one root labeled as "root" and a set of item prefix sub-trees as the children of the root, and a frequent item header table. Each node in the item prefix sub-tree consists of three fields: item-name, count, and node-ink, in which item-name registers which item this node represents, count registers the number of transactions represented by the portion of the path reaching this node, and node-link links to the next node in the FP tree carrying the same item-name, or null if there is none. Each entry in the frequent-item header table consists of two fields, (1) item-name and (2) head of node link, which points to the first node in the FP-tree carrying the item-name. FP-Growth is divided into three steps.

Step 1: Database is fetched and then count of items is taken. On the basis of minimum support threshold, frequent items are selected and sorted.

Step 2: FP tree is then initialized. From the frequent items a node list is created which will be connected to nodes of the tree. After initialization the database is fetched again. This time, item is added to the tree structure, if an item in a transaction is selected as frequent

Step 3: Starting from the least frequent item, a frequent pattern finder procedure is called recursively. If the support count of the patterns is found to be frequent, then they are displayed.

## 3. Hiding Sensitive Association Rules using EMDSRRC

Association rule hiding problem can be stated as: converting the original database into sanitized database. Sanitized database is a database on which mining on private rules is not possible only non private rules will be available. A problem can be defined as:

Let D be the traditional (original) database, R be set of association rules and subset S be set of private rules which database owner want to hide. And D' is sanitized database. So problem is to find D' such that when mining is done on D',all private rules in S will be hidden and all non private rules can be mined.
The reason behind association rule hiding is to satisfy the following conditions

1. Private database must not reveal any private rules.
2. Private database must facilitate mining of all non private rules.
3. Private database must not generate any new rules, not present in D.

The problem of finding an optimized private database, which satisfies all these conditions, has been proved as NP-hard. In existing algorithm MDSRRC, Apriori algorithm is used for generating rules in which candidate set generation is costly, when there are a large number of patterns and/or long patterns exist[8]. So it reduces the performance of MDSRRC. Hence to overcome this limitation, EMDSRRC is proposed where FP growth algorithm is used to generate rules in which divide and conquer approach is used and it is more efficient than Apriori algorithm and is also a time and efficient gives better performance[7][8].

Proposed algorithm hides rules with multiple items in L.H.S and multiple data items in R.H.S. So the rule is like aX→bY. Here b is an item selected by proposed algorithm in order to reduce the support of the R.H.S. and reduce the confidence of the rule below MCT. 1 is replaced to 0 in some transaction to decrease the support of selected items.

Some essential definitions of terms are used in the algorithm is as follow:

1. Sensitivity of Item: is number of private rules which contain this item.
2. Sensitivity of Transaction: is the total of sensitivities of all private items which are inclusive in that transaction.

The algorithm is starts with mining the association rule from the original database D using association rule mining algorithm e.g. FP growth algorithm. Then user specifies some rules as private rules (S) from the rules generated. Then algorithm counts occurrences of each item in R.H.S of private rules. Now algorithm detects IS(Set of items present in consequent of private rules with decreasing order of their frequency in consequent of private rules)= {is0, is1… isk } k≤n, by arranging those items in decreasing order of their counts. After that sensitivity of each item is calculated then sensitivity of each transaction is calculated. Then transactions which support is0 are sorted in descending order of their sensitivities.
Now hiding rule is started by selecting first transaction from the sorted transactions with high sensitivity, item is0 is deleted from that transaction. Then support and confidence of all private rules is updated and if any rules have support and confidence below MST (minimum support threshold) and MCT (minimum confidence threshold) respectively then it is deleted from S. Then sensitivity of each item, transaction and IS is updated. Again transaction with higher sensitivity is selected and is0 is deleted from it. This process is carried on until all private rules are hidden[10]. In this way, modified transactions are updated in the original database and new database (sanitized) is generated D'.Proposed algorithm MDSR1RC is used to hide the private rules from database. Given a database D, MCT and MST algorithm generates sanitized database D'. Sanitized database hides all private rules and maintains data quality[5].

1. Apply FP growth on given database D. Generate all possible association rules R.
2. Select set of rules S as private rules, Calculate sensitivity of each item j & Calculate sensitivity of each Transaction.
3. Count occurrences of each item in R.H.S of private rules. Select the transactions which supports is0, then sort them in descending order of their sensitivity. If two transactions have same sensitivity then sort those in ascending order of their length.
4. While(S is not empty)
5. Start with first transaction from sorted transactions,

6. Delete item is0 from that transaction.
7. For each rule S
8. Update support and confidence of the rule r.
9. If(support of r < MST or confidence of r < MCT)
10. Delete Rule r from S and Update sensitivity of each item.
11. Update IS (This may change is0).
12. Update the sensitivity of each transaction.
13. Select the transactions which are supports is0,
14. Sort those in descending order of their sensitivity.
15. Else take subsequent transaction from sorted transactions, go to step .
16. End

In this way, EMDSRRC algorithm used FP growth algorithm for association rules generation which performs better than MDSRRC which uses Apriori algorithm. FP growth preserves complete information for frequent pattern mining and never breaks a long pattern of any transaction. Also it reduces irrelevant information that is infrequent items are ignored. The FP growth algorithm solves the problem of finding lengthy frequent patterns to searching for the smaller ones recursively and then after adding the suffix. It uses the least frequent items as a suffix which offers excellent selectivity.

## 4. Comparative Result

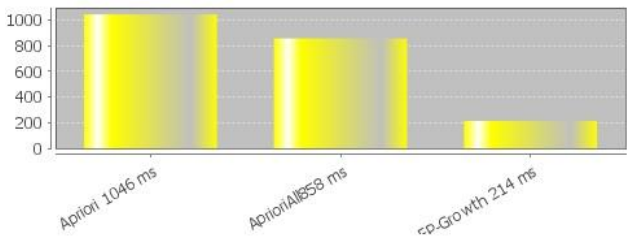


**Figure 1:** Comparative Result of MDSRRC and EMDSRRC

In proposed algorithm EMDSRRC, we have implemented FP growth algorithm to generate rules and hide sensitive rules accordingly. And we have also implemented Apriori algorithm generate rules for comparison with FP growth algorithm So Apriori and FP growth algorithms are applied on database check the efficiency. Here log file of a web browser was ven as input to both the algorithms. And time taken by both the algorithms is shown in Fig. 1 shows that FP growth algorithm is efficient than Apriori as it takes less time to generate rules and hide private rules to generate sanitized database.

## 5. Conclusion

In this way, we have implemented EMDSRRC algorithm, in which private rules are hidden to maintain privacy and quality data. In this paper, we have generated association rules using FP growth algorithm which is time efficient than Apriori Algorithm. Result shows that proposed EMDSRRC which uses FP growth algorithm to generate rules and consequently hide sensitive rules to generate sanitized database is more time efficient than MDSRRC which used Apriori algorithm. FP growth uses FP tree to generate rules. And then sanitized database is created in which private rules are hidden. This done by updating support and confidence of private rules which are marked as per requirement.

## References

[1] V.S. Verykio, A.K. Elmagarmid,E. Bertino,Y. Saygın4, and F. asseni."Association Rule Hiding",IEEE, Volume:16 Issue:4, 2003

[2] Komal Shah, Amit Thakkar, Amit Ganatra, "A study on Association Rule Hiding Approaches" International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-1, Issue-3,2012

[3] M.AtallahE.Bertino,A.Elmagarmid, M. Ibrahim,V.Verykios."Disclosure Limitation of Sensitive Rules",IEEE,1999

[4] Yitao Duan and John Canny, Justin Zhan," Efficient Privacy-Preserving Association Rule Mining" P4P Style,IEEE, 1-4244-0705-2,2007

[5] Nikunj H. Domadiy, Udai Pratap Rao, "Hiding Sensitive Association Rules to Maintain Privacy and Data Quality in Database", IEEE transactions on knowledge and data engineering,2013

[6] C. N. Modi, U. P. Rao, and D. R. Patel, "Maintaining privacy and dataquality in privacy preserving association rule mining," 2010 Second Iternational conference on Computing, Communication and Networking Technologies, pp. 1‐6, Jul. 2010.

[7] Rahul Mishra, Abha choubey, "Comparative Analysis of Apriori Algorithm and Frequent Pattern Algorithm for Frequent Pattern Mining in Web Log Data",IJCSIT,ISSN:0975-9646,2012

[8] Dr. Kanwal Garg ,Deepak Kumar,"Comparing the Performance of Frequent Pattern Mining Algorithms", International Journal of Computer Applications,0975-8887,2013

[9] S.-L. Wang, B. Parikh, and A. Jafari, "Hiding informative association rule sets," Expert Systems with Applications, vol. 33, no. 2, pp. 316 –323, 2007

[10] Y.-H. Wu, C.-M. Chiang, and A. L. Chen, "Hiding sensitive association rules with limited side effects," IEEE Transactions on Knowledge and Data Engineering, vol. 19, pp. 29–42, 2007.

## Author Profile

**Marathe Shashank S.** received his Bachelors degree B.E. in Computer Engineering from Sinhgad COE, Pune in 2012, now pursuing Master of Engineering (ME) in Computer Engineering from Savitribai Phule Pune University, Pune.

**Manjusha Yeola** is an Assistant Professor in Department of Computer Engineering, Alard College of Engineering and management, Pune, Savitribai Phule Pune University, Pune. She has over 10 years teaching experience.