# Presenting a Hiding Algorithm for Improving Privacy Preserving in Association Rule Mining

Somayeh Ghiasi,
Iran, Esfahan, Sheikhbahaee University, School of Engineering

Mehdi Bateni,
Iran, Esfahan, Sheikhbahaee University, School Of Engineering

## ABSTRACT
Association rules mining is one of data mining techniques to extract useful patterns in the framework of the law. The major problem of this technique on a database of sensitive information is disclosed to the security and privacy risks. One of the most effective solutions for maintaining privacy in data mining techniques to hide a lot of elements sensitive ( sensitive frequent patterns) reserved. In this study, an algorithm to hide the sensitive rules based on the rules and techniques to support the reduction of turbulence is presented. The proposed algorithm is to select the most appropriate transaction for changes, consider the degree of overlap between their response elements. Critical element in choosing transactions to correct the Frequency sensitive elements in the sensitive patterns and frequency patterns insensitive response elements in the balance. The proposed algorithm with algorithms ADSRRC, SIF-IDF and SL-HS dense and non-dense on four databases are implemented. Since the implementation of the proposed method compared with other algorithms reduced. Also, the number of missing rules changes the rules of the bogus transactions and the proposed algorithm is more efficient than other algorithms.

**Key Words— privacy, hiding the sensitive rules, Data Mining, secure database.**

## 1. INTRODUCTION
Data is the heart of the business process, most firms are considered, regardless of the type of industry in all small and large scale industries such as retail, telecommunications, manufacturing, utilities, transportation, insurance, credit cards and banking interactions operating systems are formed. In this context, data mining knowledge, knowledge is one of dozens of extraordinarily rapid expansion in recent years has been in the world.

Frequent patterns, which are patterns in transactional databases is greater than a certain threshold occurred. There are many different types of patterns, such as patterns of abundant elements, patterns and pattern languages abundant sequences abundant infrastructure. Common elements of the set of elements that form frequently occur together in a transaction database.

For example, customers in a shopping cart review, we can know how many items were bought together, and set the aforementioned goods as an example for the goods on the shelf in a store not is viewed as order. Mining frequent patterns lead to the discovery of dependencies between components in large datasets can be transactional or relationship.

Attalla [16] first issue of frequent pattern hides and proposed an association rule. The authors of this paper are stated immunization database to hide the sensitive frequent patterns and as well as adverse effects on the primary database and insensitive frequent patterns are also minimized.

Methodologies hide many patterns follow the following objectives:

- The frequency of visual patterns that are sensitive to the owner of the database and the minimum support levels adhere to the original database shall be the minimum amount of support or additional immunization is extracted from the database.
- All frequent patterns which are extracted from the original database minimum support, should be the same as or greater than the minimum amount of support from the extracted immunization database.
- Models have the following minimum level of support cannot be extracted from the original database, not in the least amount of support or greater than the base-data immunizing be extracted.

So many patterns to hide the side effects are as follows [17]:

- Hide failure rate in many patterns: the number of frequent patterns, since the process of hiding sensitive, hidden and safe-making was extracted from the database.
- The abundance patterns of missing: the number of frequent patterns is insensitive to the process of hiding sensitive frequent patterns extracted from a database of safe-fail.
- The frequent patterns of idealized: the number of frequent patterns suggests that the primary database cannot be extracted by mining the database is secure.

Data mining techniques to extract useful patterns from large amounts of data and knowledge implies. One of the most frequently used data mining techniques to extract the rules of the forum. Data mining techniques, in addition to their advantages confidentiality of private information contained in the original database into harm's way. To deal with the issue of privacy in the data mining techniques on data mining was an important issue. Some of privacy in the data mining techniques, techniques for hide sensitive association rules and frequent patterns is based on reducing the amount of support. The main focus of this research is on the algorithms.

## 2. RELATED WORKS
In this section some of the previous studies on privacy in mining frequent patterns and association rules are studied and assessed in well.

In 1999, M. Atallah and colleagues [16] first practical algorithm for hiding sensitive association rules from frequent patterns by reducing the amount of support manufacturer-

sensitive rules set by the user is less than the minimum amount of support offered. This algorithm reduces the amount of support is so sensitive rules to be less than the specified minimum support and attention that is to become law. That will preserve the privacy of sensitive rules. Solutions such as limiting access to the original data, the change in the original database and the publication of the proposed sample rather than the entire database, they are based on a heuristic approach. Heuristic approaches, such algorithms are scalable and effective in hide so that a series of transactions to selectively alter a database. The main strategy of this approach is blocking and turbulent [19]. Heuristic approach does not guarantee a global optimal solution, but a solution is close to one of the best solutions and gives faster response time [18].In this paper, the authors prove that the problem of finding the optimal immunization database is NP hardness. For evaluation, the algorithm was compared with the rotational algorithm of frequent patterns insensitive remaining criteria were evaluated. The algorithm is sensitive to input frequent patterns are hidden. Algorithm, the model based on the amount of support they will be sorted in descending order. The first model is sensitive to the selection list, and attempt to hide it. The algorithm is sensitive patterns are hidden one at that. After finishing each time the algorithm is run, it checks the list of sensitive frequent patterns. If you have a sensitive pattern hiding removes it from the list.

In 2001, E. Dasseni and colleagues [20] Tuesday empirical algorithm for hiding sensitive association rules based on the reduced support and confidence rules, but not both argue that every single time a rule, can be hidden. The first algorithm is sensitive rules ensure preceded by growing support base until they make less than minimum confidence the rules reduce the rule as long as safety rules or by less the reduction of support at least ahead to make sure. The third algorithm, a sensitive rule is supported by the reduced support either the front or the result of the reduction rule until the rule is less than the minimum support and confidence or support base will make less than minimum. In this paper, for simplicity, assume that algorithms and methods such as:

- hiding sensitive rules should not only be based on the support or the confidence of both must be done.
- At any moment can be a sensitive rule is hidden. No more.
- Each time the algorithm is run, the amount of support or the confidence is only one unit can be deducted.
- Criterion of reducing the values of support and confidence, decrease the side effects on the rules is insensitive. There is no overlap between sensitive rules necessary condition for this algorithm can be considered as a flaw.

In 2001 Saygin, Verykios and Clifton] 21 [first scholar who used to hide sensitive association rules instead become an unknown quantity to zero and zero to one, were proposed. The techniques described in this work for the applications used in the amounts of characteristics or confidential or unavailable. So instead these values are uncertain, such as to minimize adverse effects on rules insensitive. In this work, the authors suggested that the first three methods are simple heuristics to reduce the amount of support and the other two to reduce the safety rules are dependent on the sensitive associative. The solution for applications that do not produce erroneous rules because of incorrect rules can have harmful results. These algorithms are implemented in terms of the total number of missing rules and the new rules are generated and compared. These algorithms have unrealistic rules and regulations are missing.

In the year 2003, SRM Oliveira and colleagues [22] presented an algorithm called SWA that regardless of database size and the number of sensitive rules that should be hidden need scan once database. The algorithm is based on memory, so it can be used on very large databases. For each rule, the method is sensitive in revealing a defined threshold. There threshold for each sensitive rule, creating a balance between the hiding-making rules is insensitive sensitive detection rules.

Here is the database owner must determine sensitive rules. In the algorithm, the transaction is characterized by sensitive and resistant. The current transaction is one element that is most frequent among the patterns is designated as critical elements to be considered a victim. Transactions that need to be corrected and transaction support specified for each model are arranged in ascending order. Transactions to hide the sensitive rules are altered. The disadvantage, however, is the failure rate in hiding associative rules.

In 2005, SRM Oliveira and colleagues [23] several methods for hiding sensitive rules were introduced simultaneously. This algorithm regardless of the number of critical elements, the database is scanned twice to apply. In the first scan, to increase transaction speed by a sensitive index file will be created in the next scan, the primary database for securing privacy and hide sensitive rules, paid. Algorithms are MinFIA, MaxFIA, IGA and Naive. In this paper cloak failure rate criteria, the rules are missing, the pattern and degree of dissimilarity were considered unrealistic. Cloak failure rate is a measure of not hiding sensitive rules after immunization database states. The measure is the percentage of missing rules hiding non-sensitive rules (which should be hidden) after immunization of database states.

In 2007, B. Parikh and colleagues [24] for hiding predictive association rules (the rules that contain sensitive elements on the front are) two algorithms are presented. Both algorithms automatically hide this rules. To identify sensitive rules that should be hidden sensitive rules need data mining and manual selection process before they are hiding. The first algorithm, called ISL By increasing support of sensitive elements left rule, the rule will reduce reliability. The second algorithm called DSR by increasing support for the right set of rules, which reduces confidence in the rule. DSR algorithm is much more support for those who have sensitive elements, it is efficient.

The output of the algorithm in order to eliminate the dependence of the critical elements and different database from the disadvantage is secure.

In 2010, Modi, CN and colleagues [25] have introduced a clustering algorithm called DSRRC based on the common law right rules, the rules simultaneously as possible and with minimal changes to the database to hide.

In this method, transactions are sorted in descending order of their sensitivity. The most sensitive transactions has decreased below the threshold level of confidence confidently rule, will be changed. To reduce the level of confidence in the rule, the rule will be reduced by the support element on the right. To reduce the amount of the support element of the right rule, the critical element in the transaction becomes zero.

In the year 2012, Dhyanendra Jain and colleagues [26] proposed an approach based on optimization techniques in which turbulent changes and no change of position sensitive items on the support items are concerned. Also, the size of the database will remain unchanged and will apply any increase and decrease transaction. The advantage of the algorithm proposed in this paper, the non-sensitive items, and the failure to support the database and hide the maximum number of rules in transaction databases with minimal changes to the. In this paper, two general common strategies to prevent misuse of data have been proposed. The data before data mining, and other publications, only a subset of the full data using a distributed database, is.

In 2012, in another article Komal Shah and colleagues [27] algorithm corrects DSRRC and called it their ADSRRC. The article stated that the disadvantage of the algorithm is to choose the transaction DSRRC to change, depending on the transactions in the database. To overcome the disadvantage of starting-solution state that such transactions will be sorted in descending order of sensitivity and length. Moreover DSRRC algorithm after each change of a transaction, the transaction sorting operation based on their sensitivity, but allows this sort of algorithm operation transactions are carried out only once ADSRRC takes. These algorithms are also hiding sensitive rules; the number of rules is also more missing. In addition, this paper proposed a new algorithm called RRLR rules that are more elements in their own right cannot hide. In this algorithm to hide the sensitive rules are also supported and can be safely reduced.

In 2013, Hai Quoc Le and colleagues [18] to hide the sensitive rules to manage the risk of exposing sensitive data to be shared when they began. this study of a new heuristic algorithm to hide the offers. This heuristic algorithm is based on three steps:

- Determine the victims modify the item so that it can be taken to minimize the impact on frequent item sets.
- Specify the minimum number of transactions that need to be modified, so as to reduce distortions in published databases.
- Remove items from the transactions identified victims so that they maintain frequent item sets.

In accordance with previous research, the limitations and drawbacks of the algorithms proposed by researchers there. For example, the algorithm presented by E. Dasseni and colleagues [20], because there is no limit to the overlap in the sensitive rules is introduced. In both algorithms, ISL and DSR, the database may be due to the removal of safety-critical elements vary. Limitations of the algorithm DSRRC, this is only one element in their own right are rules that have to hide it. Selection of affiliated transactions (for a change) to arrange the transactions in the database, the disadvantage of this method is considered. ADSRRC algorithm, the number of missing rules too much.

## 3. PROPOSED METHOD

The proposed algorithm is a heuristic based method is skewed, thus eliminating a critical element of the selected victim transactions; sensitive patterns reduced the amount of support. Support removal process for reducing the amount of sensitive patterns are repeated until the amount of support transaction-sensitive patterns selected from the support threshold (MST) is minimal and hide sensitive patterns. The proposed algorithm, the algorithm is based on the amount of support. This approach is sensitive patterns must be specified by the user.

In this algorithm, transaction-sensitive non-critical transactions have been isolated and studied their properties. Transactions are crucial for improving the selection of the most sensitive overlay pattern is. If two or more transactions, sensitive, have the same level of support, then choose the most appropriate balance between the frequency of transactions with a transaction-sensitive response elements in patterns and frequency insensitive patterns transaction will take place. Select the transaction with maximum overlap makes securing databases with minimal changes may be made.

The proposed algorithm is discussed in the following concepts:

- Per Transaction: Number of elements in the transaction.
- The model: the number of elements in the template.
- Sensitive transaction: A transaction includes at least one pattern is critical.
- Transaction Overlap: there is a common element between the patterns of interaction are critical. The number of common response elements sensitive to patterns in most transactions, the amount of overlap in the transaction will be.
- Fs: Frequency sensitive elements in delicate patterns.
- Fns: Frequency sensitive elements in non-sensitive patterns.
- Balancing factor (a): a measure to create a balance between the values of Fs and Fns.
- Equilibrium abundance of elements: the value of this parameter is calculated using equation 1.
- Equilibrium frequency of transactions: Total balance frequency sensitive elements of the transaction.
- Victim element: the element being selected to secure database transactions is eliminated.

To select an element in the transaction must be chosen victim-sensitive patterns in the transaction is registered. The most common patterns are sensitive.

Select the element with the highest frequency of sensitive patterns could decrease the amount of change to secure database transactions. Element in the transaction as victim element is chosen to be a member of sensitive patterns in the transactions and non-sensitization patterns have the lowest rate. Select the element with the lowest rate of non-sensitive patterns reduces the number of patterns in a secure database.

So you must choose the best elements of the victim's most sensitive and least frequent patterns in non-sensitive patterns is balance. Using equation 4.1 we can create this balance.

$$FB = (a * Fs) + ((1-a) * (Max (Fns) - Fns)) \quad \text{equation (1)}$$

Balance factor (a) is determined by the database administrator. This value is consistent with the purpose of sharing the database manager set. Interval is a value between 0 and 1 is chosen. If a value smaller than 0.5 is considered, the number of missing rules to the amount of change in a secure database with a higher priority transaction is.

If a value greater than 0.5 is considered, reducing the transaction changes to the rules of priority is lost in a secure database with. The last element is selected as the victim has the highest FB. If several elements have equal amounts of FB element is part of the pattern length is more critical.

Details of the algorithm are as follows:

Input: set of transaction data D = {T1, T2,..., Ti,..., Tn}, the threshold level of support for the MST by the user specified patterns and sensitivity S = {si1, si2,..., sij,..., sin} this set is determined by the owner of the database.

Output is a secure database so that the sensitive patterns derived from them will not be lost and the least number of non-critical patterns.

- Step one: calculate the following values:

  - Frequency sensitive elements sensitive patterns: the number of occurrences of each critical element of the sensitive patterns calculated in descending order.

  - Frequency sensitive elements in non-sensitive patterns: the number of occurrences of each element in sensitive, non-sensitive patterns are calculated and arranged in ascending order.

  - Transaction Sensitivity: Total length of critical patterns in a transaction.

  - The Transaction Overlap: number of common response elements sensitive patterns in the transaction.

  - List of sensitive patterns for each transaction: all sensitive patterns in the transaction.

  - Number of delicate pattern matching: Number of delicate patterns in the transaction.

  - largely overlapping sensitive transactions are calculated patterns. Overlaps between sensitive patterns, there is a critical element or elements in a pattern too sensitive to be shared in a transaction.

- Step Two: Select the transaction

  - Transaction Tb is the maximum amount of overlap in their selection. If two or more than two transactions are critical transactions, sensitive, had the highest amount of overlap, then:

  - FB value for the transactions and non-sensitization patterns and calculated transaction that is highest is chosen. If the FB has more than one maximum length is less than the transaction is selected.

- Step Three: Select the element of sacrifice

  - FB for all critical elements of the transaction Tb is calculated in descending order. Element that FB has the highest value is selected as the victim. If FB is more than one element has a maximum value when the element is among the most sensitive pattern are selected.

- Step Four: Remove the victim element

  - Transaction Tb victim element is eliminated A single element of the support of all the models that have been deducted from the minor victim and quantities updated. If the support of the MST model is less sensitive to the delicate pattern of sensitive patterns be removed and the update frequency sensitive elements.

- Second to fourth steps are repeated until the set is empty sensitive patterns.

To better understand the delicate balance between the abundance of sensitive patterns and frequency sensitive elements in non-sensitive patterns, Example 1 is given.

Example 1: A series of sensitive elements S = {a, b, c, d, e} is given. It is assumed that the frequency of the critical elements in the sensitive patterns (Fs) and frequency-sensitive elements in non-sensitive patterns (Fns) is shown in Table.1:

**Table.1: Frequency sensitive elements in sensitive patterns and patterns insensitive.**

| | $F_s$ | $F_{ns}$ |
|---|---|---|
| **A** | 5 | 5 |
| **B** | 4 | 2 |
| **C** | 4 | 4 |
| **D** | 2 | 5 |
| **E** | 2 | 4 |

FB values for different values of Balance factor 0.5, 0.25, and 0.75, respectively, calculated using equation 1 as shown in Table.2.

**Table.2: Bf values are sensitive elements for different values of Balance factor**

| | $F_s$ | $F_{ns}$ | a= 0.5 | a= 0.25 | a= 0.75 |
|---|---|---|---|---|---|
| **A** | 5 | 5 | 2. 5 | 1. 25 | 3. 75 |
| **B** | 4 | 2 | 3. 5 | 3. 25 | 3. 75 |
| **C** | 4 | 4 | 2. 5 | 1. 75 | 3. 25 |
| **D** | 2 | 5 | 1 | 0.5 | 1. 5 |
| **E** | 2 | 4 | 1 | 1. 25 | 1. 75 |

For example, if the critical elements b and c, the elements are candidates to choose victim element coefficient value of 0.25 were assumed to be a balance, an appropriate sacrifice to be selected as follows:

For a =0.25 (reduction rules to reduce variations Missing Transaction priority):

FBb = (0.25 × 4) + ((1-0.25) * (5-2)) = 3.25

FBc = (0.25 × 4) + ((1-0.25) * (5-4)) = 1.75

Since the value of the equilibrium factor of 0.25 is assumed, the critical elements listed candidates shall be selected element as an element of sacrifice that is primarily non-sensitive patterns has the least impact on the patterns of missing reduced. Secondary to the reduction of transaction changes the result. According to the data in Table.2 and compared the amounts of FBc and FBb, b element that is much more FBb (FBb> FBc) as a sacrificial element is selected. As we have seen lots sensing element b in non-sensitive patterns is less sensitive element c. By removing the critical elements that occur less frequently in non-sensitive patterns, the less impact there will be on non-critical patterns of missing data reduced the number of rules.

If the list of candidates for the sensitive elements S = {b, c, d} is the equilibrium coefficient of 0.75 was assumed to be a value, select an appropriate victim is chosen as follows:

For a =0.75 (missing rules change mitigation to reduce transaction priority):

FBd = (0.75 × 2) + ((1-0.75) * (5-5)) = 1.5

FBc = (0.75 × 4) + ((1-0.75) * (5-4)) = 3.25

FBb = (0.75 × 4) + ((1-0.75) * (5-2)) = 3.75

According to calculations FBb> FBc> FBd b element as a result of the victim element is selected. According to Table.2, it is observed that the sensitive frequency sensing element patterns with frequency sensitive element b c d is equal to the frequency sensitive element (Fs b = Fs c> Fs d) and b element abundance patterns insensitive comparison of two elements: less is more (Fns c < Fns d < Fns b).

The purpose of the database owner of balance factor of sharing can be adjusted.

Transactions between changes and reduce the number of missing rules are maintained in secure databases.

## 4. COMPARE PROPOSED METHOD WITH EXISTING METHODS

In this chapter the results of experiments performed on a database Mushroom, Web View1, Web View2 and Chess on algorithms SIF-IDF, ADSRRC, SL-HS [29] and the proposed algorithm is shown. Runtime evaluation criteria, the number of frequent patterns of missing data, the number of dummy patterns and the total number of transactions are eliminated.

The proposed algorithm and other algorithms have been implemented in C # on a PC with 512MB main memory and processor specifications Pentium (R) -Duel 7 operating system is running at a rate of two GHz. Data Mining frequent patterns to obtain the Weka data mining software is used. Then the frequent patterns extracted from the dataset for the Mushroom dataset, 3, 5, 7 Template for Chess Dataset 4, 6 and 9 model for the Dataset Web View1, 20, 40, 60, 80, 100 View2 Web Template for data collection, 118, 234, 350, 446 and 582 model as the selected tender. To calculate the number of frequent patterns generated from database FPGrowth algorithm is used. In these experiments, the algorithm proposed by SIF-IDF, ADSRRC and SL-HS compared to the number of missing rules, the number of spurious rules, runtime and number of deleted elements is examined.

View the data sets used in the experiments is shown in Table.3. The number of frequent patterns extracted from the support threshold before immunization data set is shown in Table.4.

**Table.3 specification Database**

| Average length transaction | Maximum length transaction | Number of Elements | Number of transactions | Database |
|---|---|---|---|---|
| 23 | 23 | 119 | 8124 | **Mushroom** |
| 5 .2 | 267 | 497 | 59602 | **Web-View1** |
| 5 | 161 | 3340 | 77512 | **Web-View2** |
| 37 | 37 | 75 | 3195 | **Chess** |

**Table.4: Number of frequent patterns extracted according to the specified MST**

| Number of frequent patterns | Support threshold | Database |
|---|---|---|
| 565 | %40 | **Mushroom** |
| 808 | %0.2 | **Web-View1** |
| 3683 | %0.2 | **Web-View2** |
| 413 | %40 | **Chess** |

**Criteria for the evaluation of the proposed algorithm:**

- Runtime
- The number of frequent patterns of missing
- The number of elements removed (rate of change in the database)
- The number of bogus rules

The results of the proposed algorithm. The efficiency of the proposed algorithm in terms of non-sensitive frequent patterns of missing and deleted from the total number of transactions in hiding sensitive frequent patterns are evaluated.

Specifications the data sets used in these experiments are shown in Table.3. Algorithm in terms of efficiency with ADSRRC, SIF-IDF and SL-HS compared. Since the implementation of the proposed algorithm compared to the algorithm of SIF-IDF, SL-HS and ADSRRC improved by reducing the number of elements has been removed because it supports transactions and decrease the sensitivity of the model simultaneously, the algorithm can dramatically reduce finds.

Fig.1, to improve the performance of the proposed algorithm over other algorithms for minimum support threshold is 40% for the database Mushroom show.

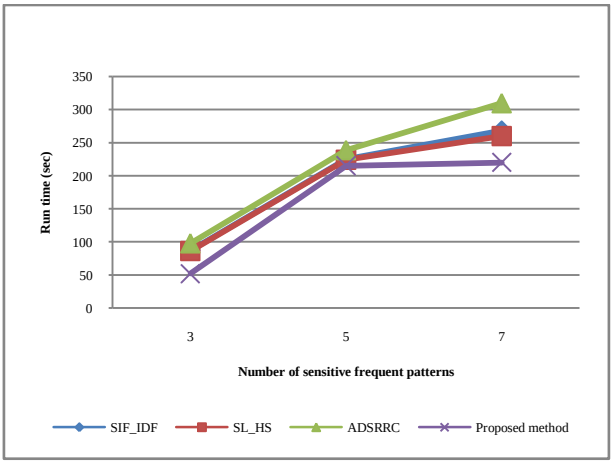


**Fig.1: Comparison of runtime support for a minimum of 40% for Mushroom**

In Fig.2 the proposed algorithm reduces the number of missing patterns than other algorithms for minimum support of 40% for the database Mushroom shown .

Select the most sensitive transactions and reduce overlapping transactions and sensitive to changes in the balance between reducing transaction changes and the number of missing rules, missing rules of the proposed algorithm is reduced compared to other algorithms.

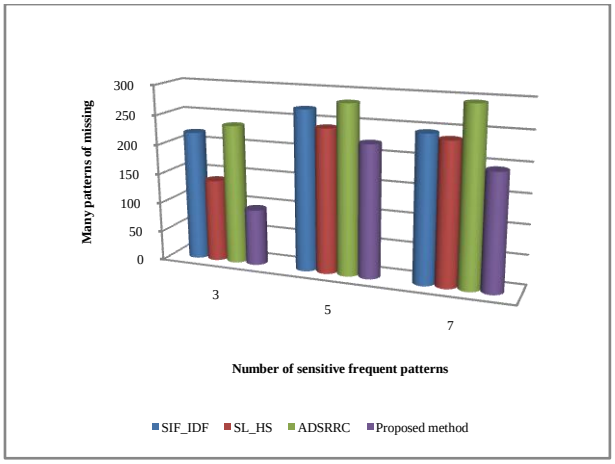


**Fig.2: Comparison of patterns of missing a lot of support for a minimum of 40% for Mushroom**

The number of missing patterns in the proposed method compared to SL-HS, ADSRRC and SIF-IDF is less. The reason for this improved balance and considering the overlap coefficient is sensitive response elements in patterns .

Fig.3 Comparison of the total number of transactions are eliminated in the proposed algorithm compared to other algorithms, for at least 40% support for database Mushroom, is shown.

Because the selection of sensitive transactions, the maximum amount of overlap measure is placed on the removal of sacrificial element of the transaction, the amount of critical support multiple patterns simultaneously reduced .

Thereby reducing transaction changes the database and reduce the number of missing rules will be followed.

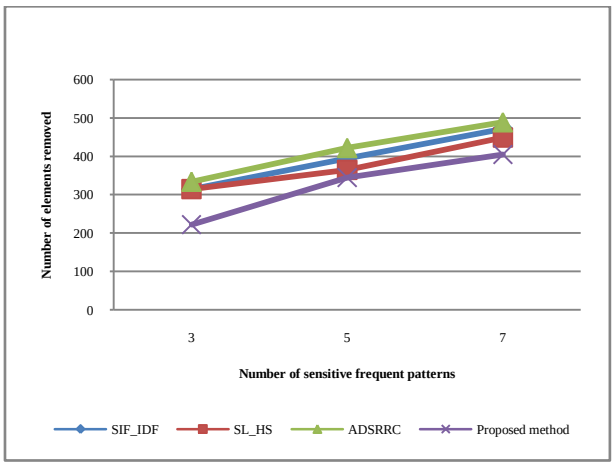


**Fig.3: Comparison of the number of elements removed from the total transactions for a minimum support of 40% for Mushroom**

Fig.4 compares the number of bogus rules in the proposed algorithm over other algorithms for minimum support of 40% for the database Mushroom shown.

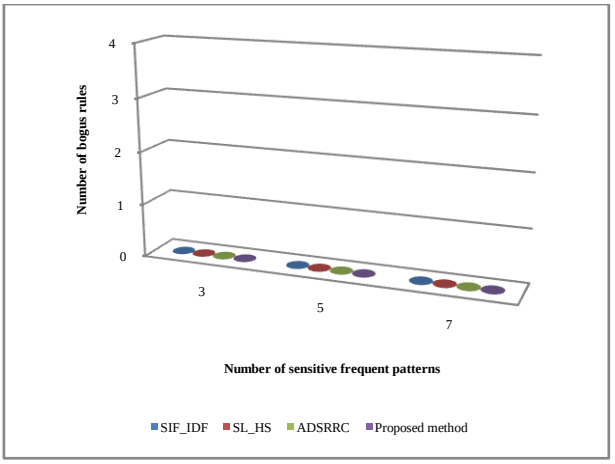


**Fig.4: Comparison of bogus rules for minimum support of 40% for Mushroom**

Fig.5 improves the performance of the proposed algorithm compared to other algorithms, for minimum support threshold is 40% for Chess DB-show displays.

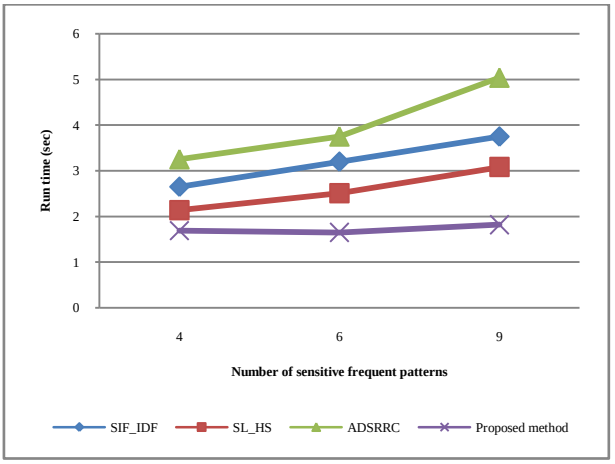


**Fig.5: Comparison of runtime support for a minimum of 40% for Chess**

In Fig.6, the number of frequent patterns missing proposed algorithm compared to other algorithms for minimum support of 40% for Chess DB is shown. With regard to the criterion of overlap between sensitivity patterns in the transaction and the balance is right, the missing rules in the proposed method compared with other reduced algorithms.

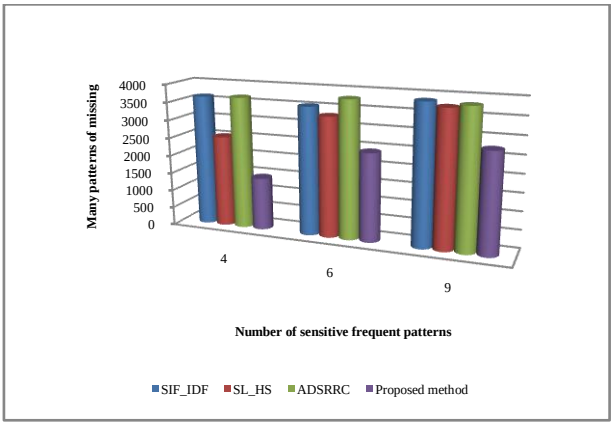


**Fig.6: Comparison of patterns of missing a lot of support for a minimum of 40% for Chess**

In Fig.7 the total number of transactions are eliminated in the proposed algorithm compared to other algorithms for minimum support of 40% for Chess DB is shown.

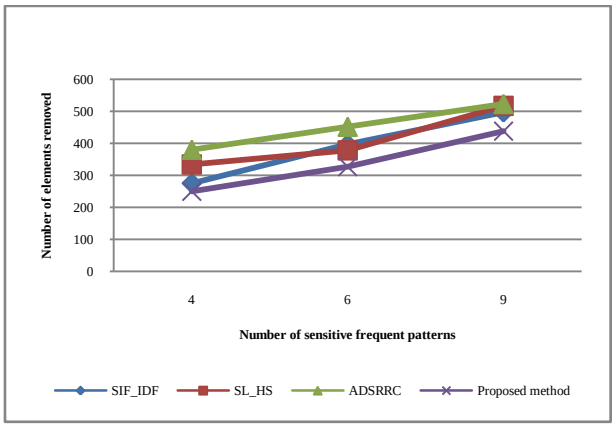


**Fig.7: Comparison of the number of elements removed from the total transactions for a minimum support of 40% for Chess**

Fig.8 compares the number of bogus rules in the proposed algorithm compared to other algorithms for minimum support of 40% for Chess DB is shown.

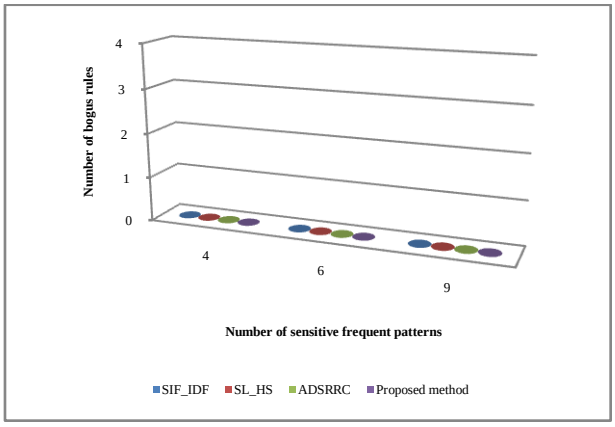


**Fig.8: Comparison of bogus rules for minimum support of 40% for Chess**

In Fig.5, 9 improved since the implementation of the proposed algorithm compared to other algorithms for the minimum support of 0.2% for the database Web-view1 shown.

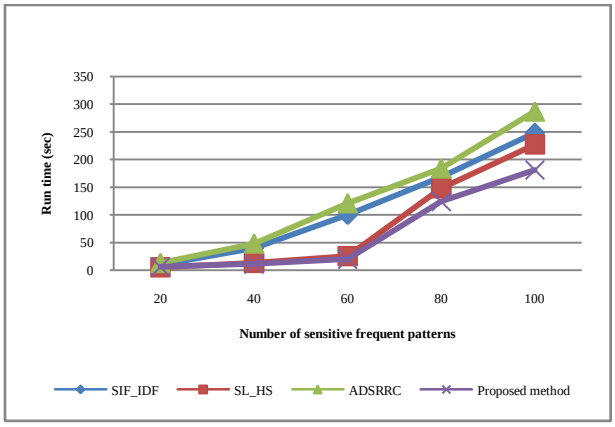


**Fig.9: Comparison of runtime support for at least 0.2% of Web-view1**

Fig.10 Comparison of the number of missing patterns than other algorithms for minimum support of 0.2% for the database Web-view1 shown. Missing rules in the proposed method compared with other reduced algorithms.

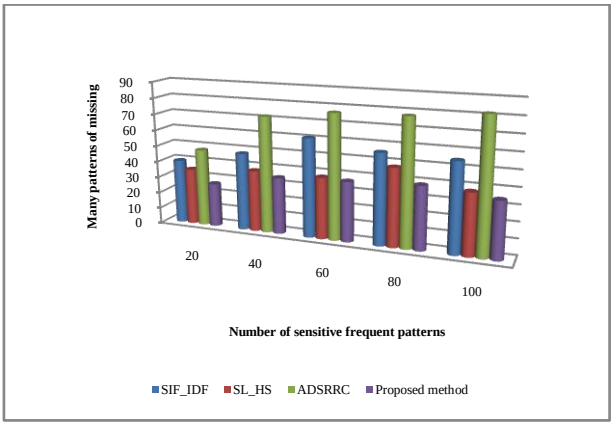


**Fig.10: Comparison of the number of frequent patterns Missing support for at least 0.2% of Web-view1**

Fig.11 Comparison of the total number of transactions are eliminated in the proposed algorithm compared to other algorithms for minimum support of 0.2% for the database Web-view1 shown.

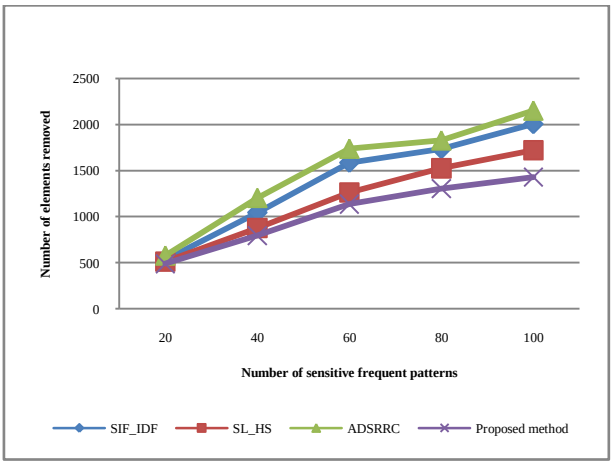


**Fig.11: Comparison of the number of elements removed from the total transactions for a minimum support of 0.2% for Web-view1**

Fig.12 compares the number of bogus rules proposed algorithm compared to other algorithms for minimum support of 0.2% for the database Web-view1 shown.

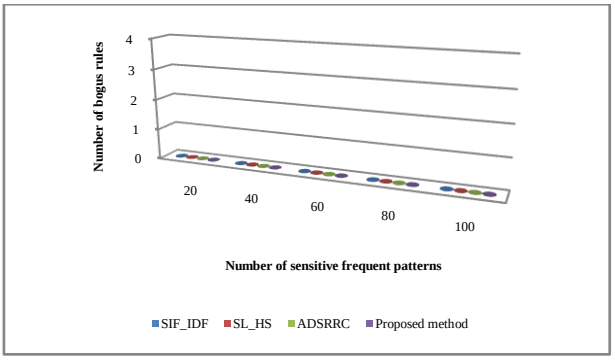


**Fig.12: Comparison of bogus rules for minimum support of 0.2% for Web-view1**

In Fig.13 time improved performance of the proposed algorithm compared to other algorithms for minimum support of 0.2% for the database Web-view2 shown.

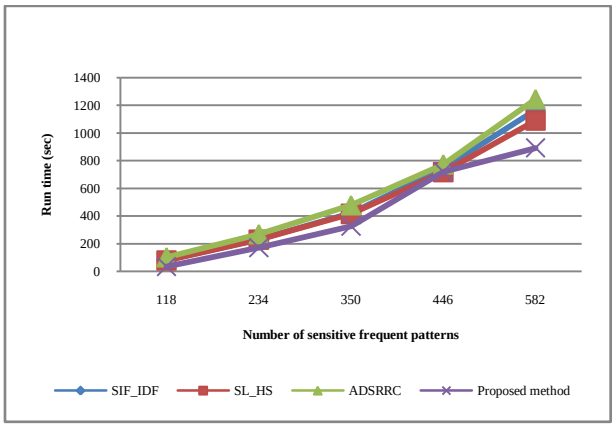


**Fig.13: Comparison of runtime support for at least 0.2% of Web-view2**

In Fig.14 patterns of missing data in the proposed algorithm compared to other algorithms for minimum support Web-view2 0.2% for the database is shown. Missing rules in the proposed method compared with other reduced algorithms.

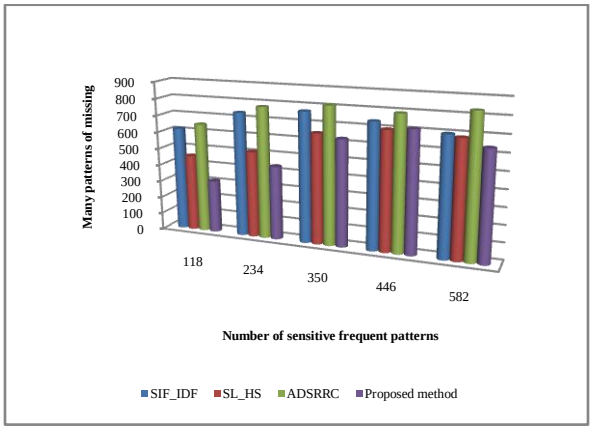


**Fig.14: Comparison of the number of frequent patterns Missing support for at least 0.2% of Web-view2**

In Fig.15 the total number of transactions is eliminated in the proposed algorithm compared to other algorithms for minimum support of 0.2% for the database Web-view2 shown.

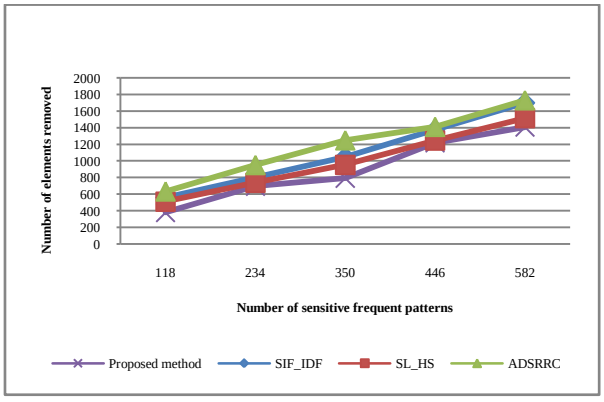


**Fig.15: Comparison of the number of elements removed from the total transactions for a minimum support of 0.2% for Web-view2**

In Fig.16 the number of rules in the proposed algorithm compared to other algorithms for minimum support of 0.2% for the database Web-view2 shown.

Web-view2

In Figures 17, 18, 19 and 20, lost a lot of patterns for different support thresholds are calculated and evaluated the proposed method over other methods.

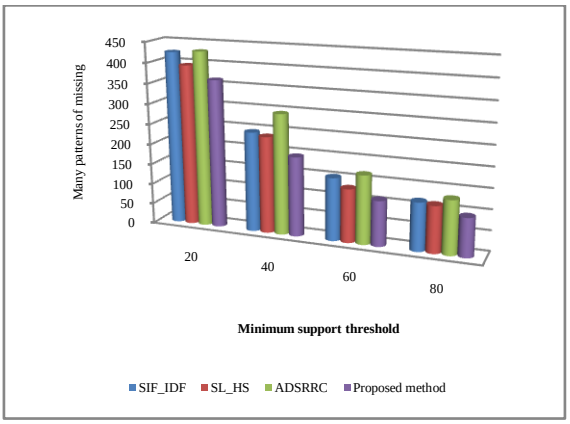


**Fig.17: Comparison of the number of frequent patterns, lost to No. 7 abundantly sensitive model for Mushroom**

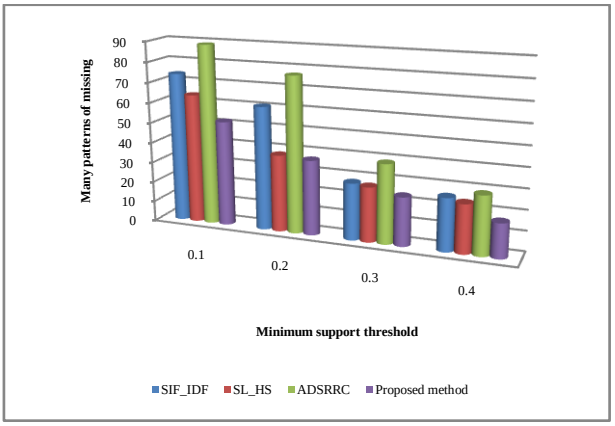


**Fig.18: Comparison of the number of lost great patterns for 60 great pattern sensitive for Web-view1**

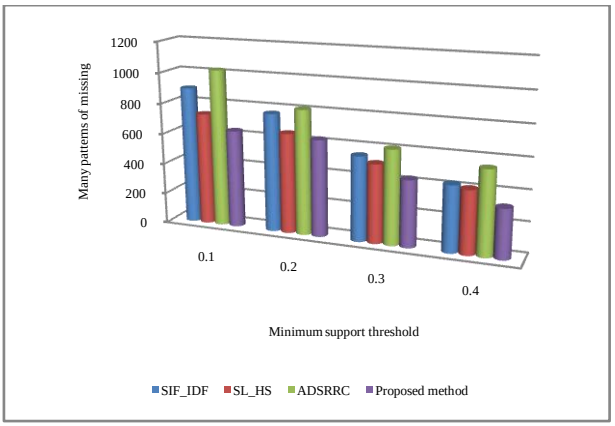


**Fig.19: Comparison of the number of frequent patterns, for a total of 350 patterns is missing a lot of critical Web-view2**

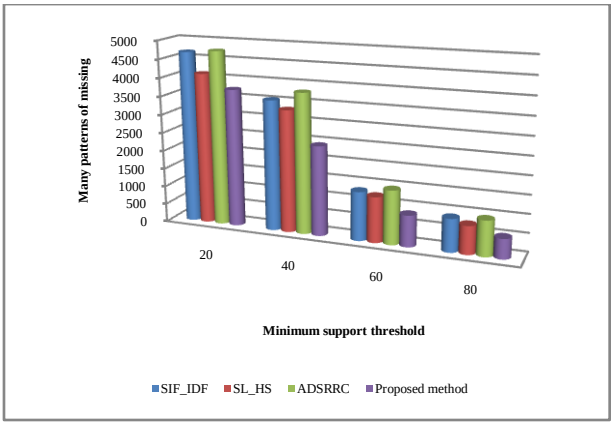


**Fig.20: Comparison of the number of frequent patterns, numerous pattern sensitive for Chess lost to No. 6**

## 5. CONCLUSIONS

Results from experiments conducted on the performance of the proposed algorithm are compared to the performance of other algorithms ADSRRC, SIF-IDF and SL-HS is as follows:

- reduce the number of frequent patterns missing proposed algorithm compared to other algorithms
- reduce the execution time of the proposed algorithm compared to other algorithms
- Reduction in the database changes in the proposed algorithm compared to other algorithms
- Complete all hide sensitive patterns in the proposed algorithm.

The results show that the proposed method over other methods in the database always has a better performance is a dense and non-dense.

## 6. RECOMMENDATIONS FOR FUTURE WORK

(1) A detailed study of algorithms, artificial intelligence and artificial intelligence techniques for association rules hiding mostly aimed at minimizing the number of missing rules.

(2) Add functionality to an algorithm that is able to calculate the level of confidence in the algorithms examined.

## 7. REFERENCES

[1] F. Rajola, "Customer Relationship Management in the Financial Industry", Springer-Verlag Berlin Heidelberg, 2013.

[2] K. N. V. D. Sarath and V. Ravi, "Association rule mining using binary particle swarmoptimization", Engineering ApplicationsofArtificial Intelligence, Vol. 26, 2013, pp. 1832–1840.

[3] D. Aruna Kumari, K. Rajasekhara Rao, and M. Suman. "Privacy Preserving Data Mining", Springer International Publishing, Vol. 249, 2014, pp. 517–524.

[4] V Garg, A.Singh, and D. Singh. "A Survey of Association Rule Hiding lgorithms", IEEE 2014 Fourth International Conference on Communication Systems and Network Technologies, Vol. 24,2014,pp. 2676 – 2678.

[5] T.P Hong, C.W Lin, K.T Yang, and S.Wang. "Using TF-IDF to hide sensitive itemsets. ", Springer Science+Business Media Appl Intell DOI 10.1007/s10489-012-0377-5,2012

[6] C-M. Wu and Y-F. Huang, "Privacy Preserving Association Rules by Using Branch-and-Bound Algorithm", Advances in Computer Science and Engineering, Vol. 141, 2012, pp. 409–416.

[7] V. Patidar, V. Shrivastava, and V. Shivastava. "A Generalized Association Rule Method for Privacy Preserving in Data Mininig", International Journal of Advanced Research in Computer Science and Software Engineering,Vol. 3,2013,pp. 703-706.

[8] X. Qi and M. Zong. "An Overview of Privacy Preserving Data Mining", Procedia Environmental Sciences, Vol. 12, 2012,pp. 1341-1347.

[9] Verykios V. S., Bertino E., Fovino I. N., Provenza L. P., Saygin, Y., and Theodoridis "State-of-the-art in privacy preserving data mining", SIGMOD Record, Vol. 33, 2004, pp. 50-57.

[10] H. jiawei, K. Micheline. "Data Mining Canceps and Technique". second edition. Elsevier Computers in Industry, Vol. 64, 2013, pp. 776–784.

[11] R. Agrawal , T. Imielinski , and Sawmi. "A Mining association rules between sets of items in large databases. " ACM SIGMOD international conference on management of data, Vol. 14,1993, pp 207–216.

[12] S. Kotsiantis and D. Kanellopoulos, "Association Rules Mining: A Recent Overview", GESTS International Transactions on Computer Science and Engineering, Vol. 32 (1),2006, pp. 71-82.

[13] K. Shah, A. Thakkarand, and A. Ganatra. "Association Rule Hiding by Heuristic Approach to Reduce Side Effects & Hide Multiple R. H. S. Items", International Journal of Computer Applications (0975 – 8887) Vol45, 2012.

[14] Elisa Bertino, Dan Lin, and Wei Jiang, "A Survey of Quantification of Privacy Preserving Data Mining Algorithms", Springer, Vol. 169, 2008, Pages: 183–205.

[15] A. Divanis, Vassilios S. Verykios "Association Rule Hiding For Data Mining" Springer, DOI 10.1007/978-1-4419-6569-1, Springer Science + Business Media, LLC 2010.

[16] M. Atallah, E. Bertino, A. Elmagarmid, M. Ibrahim, and V. S. Verykios. "Disclosure limitation of sensitive rules", Knowledge and Data Engineering Exchange, Vol. 16, 1999, pages45–52.

[17] T. -P. Hong and K. -T. Yang. "Several heuristic approaches to privvacy preserving data mining. " department of computer science and information enginieering national university of koashsiung. 2010

[18] H. Q. Le, S. Arch-int, H. X. Nguyen, and N. Arch-int, "Association rule hiding in risk management for retail supply chain collaboration", Computers in Industry, Vol. 64,2013, pp. 776–784.

[19] S. Verykios, "Association rule hiding methods", Wiley Online Library Interdisciplinary Reviews: Data Mining and Knowledge Discovery, Vol. 3, 2013, pp. 28–36.

[20] E. Dasseni, V. S. Verykios, A. K. Elmagarmid, and E. Bertino. "Hiding association rules by using confidence and support". In Proceedings of the 4th International Workshop on Information Hiding, Vol. 2137,2001, pp.369–383.

[21] Y. Saygin, V. S. Verykios, and C. W. Clifton. "Using unknowns to prevent discovery of association rules. " ACM SIGMOD Record, Vol, 30, 2001, pp: 45–54.

[22] S. R. M. Oliveira and O. R. Zaïane. "An Efficient One-Scan Sanitization For Improving The Balance Between Privacy And Knowledge Discovery", Department of Computing Science University of Alberta, 2003.

[23] S. R. M. Oliveira and O. R. Zaïane. "Privacy preserving frequent itemset mining". In Proceedings of the IEEE International Conference on Privacy, Vol. 14,2005, pp. 43–54.

[24] S-L. Wang, B. Parikh, and A. Jafari. "Hiding informative association rule sets", Expert Systems with Applications, Vol. 33, 2007, pp. 316–323.

[25] C. N. Modi, U. P. Rao,,and D. R. Patel, "Maintaining privacy and data quality in privacy preserving association rule mining", Computing Communication and Networking Technologies (ICCCNT) International Conference on,2010 , pp 1-6.

[26] D. Jain, P. Khatri, R. Soni, and B. Kumar Chaurasia, "Hiding Sensitive Association Rules without Alteringthe Support of Sensitive Item(s)", Institute for Computer Sciences, Social Informatics and Telecommunications Engineering,Vol. 84,2012,pp. 500-509.

[27] K. Shah, A. Thakkarand, and A. Ganatra. "Association Rule Hiding by Heuristic Approach to Reduce Side Effects & Hide Multiple R. H. S. Items", International Journal of Computer Applications Vol45, 2012,pp: (0975 – 8887).

[28] G. Salton, Fox EA, and H. Wu , "Extended boolean information retrieval Commun",ACM Vol. 26, 1983,pp:1022–1036.