International Journal of Database Theory and Application
Vol.9, No.7 (2016), pp.147-156
http://dx.doi.org/10.14257/ijdta.2016.9.7.13

An Efficient Method for Protecting High Utility Itemsets in Utility Mining

Anshu Chaturvedi, D. N. Goswami and Rishi Soni*

Madhav Institute of Technology and Science, Gwalior – 474005, India, Jiwaji University, Gwalior – 474011, India, Jiwaji University, Gwalior – 474011, India
anshu_chaturvedi@yahoo.co.in, goswamidn@yahoo.com, rishisoni17@gmail.com
**Corresponding author*

Abstract

Privacy preserving data mining (PPDM) has become a popular research direction in data mining. Privacy preserving data mining is an approach to develop algorithms by which we can modify the utility values of original data using some techniques in order to protect sensitive information from unauthorized user. Protecting data against illegal access becomes a serious issue when this data is required to be shared onto the network due to some reasons. To hide the sensitive information, many approaches have been proposed. In this study, we are proposing an efficient method, for protecting high utility itemsets using distortion technique where the values for high utility items are altered to achieve the privacy. Algorithm is designed in such a way so as to handle privacy without disclosure of sensitive information. The algorithm can completely hide any given utility items by scanning data iteratively. The results when compared with existing one show significant reduction in execution time.

Keywords*: Privacy preserving, Utility mining, High utility itemsets, Sanitization process*

1. Introduction

Data mining Data mining enables us to discover previously unknown and potentially useful information from huge amount of database. The knowledge discovered from this data plays an important role in decision making in the areas such as business management, marketing analysis, medical analysis, criminal records and credit records *etc.* [1]. Association rule mining is one of the common most approach in the field of data mining which determines all itemsets with support values greater than the specified threshold. To derive the utility of an itemset, utility mining came into existence which claims to be better than association rule mining in certain terms. Privacy preserving aspect of this part of data mining has gained momentum in the field of research application recently because the data set contains highly sensitive information as well and no user would like that sensitive information should be leaked to outsiders. Therefore, this sensitive information which can be mined from a database should be taken out separately, because such sensitive information can lead to compromises in data privacy when it is shared on the network. However, a key problem faced is the need to maintain the confidentiality of the disclosed data without hindering the legitimate needs of the data user. In doing so, it becomes necessary to modify the data value(s) and relationships (utility itemsets). Obtaining a true balance between the disclosure and hiding is a tricky issue [2-3]. This can be achieved largely by hiding the high utility itemsets that expose the sensitive part of the data. One such method is hiding high utility itemsets because association amongst the data is what is captured by most of the data users. Such vulnerability of high utility itemsets poses a great threat to the data if the data is in the

ISSN: 2005-4270 IJDTA

hands of a malicious user. To prevent data from being misused two common strategies are used in literature. First strategy alters the date before delivering it to the data miner [4]. Second strategy releases only a subset of the complete data using distributed databases approach.

In this study, we propose an efficient method, for protecting high utility itemsets using distortion technique where the high utility items are altered to achieve the privacy and at the same time it maintains the balance between privacy and disclosure of information. The algorithm is also compared with the existing one and it shows better results in terms of execution time.

The rest of the paper is organized as follows. The related work is presented in Section 2. Proposed work is given in Section 3 along with its description and illustrative example. Results are discussed in Section 4. Finally conclusion and future work is drawn in Section 5.

2. Related Work

There are so many PPDM techniques being established for a variety of data mining techniques such as classification, association rule discovery, utility mining and clustering which are based on the hypothesis that adaptive amendment or sanitization process and heuristics can be utilized to address the concerned issues. Utility mining is another emerging topic in the field of data mining. Identification of itemset with highest utility in terms of profit, cost, quantity *etc.* is what is known as utility mining. The process of discovering all itemsets whose utility values are either equal to or greater than a user specified threshold in a transaction database is what is known as Utility mining. Though, the utility values of itemsets do not follow the downward closure property. In other words, it can be stated that a subset of a high utility itemset may not be a high utility itemset [5-6]. Liu *et al.* [7], proposed fast alorithm for finding high utility itemsets. They discussed about mining the high utility itemsets efficiently with reduced space along with simplifying the computation complexity of utility calculation. The approach uses two phase algorithm in which phase I defines the transaction weighted downward closure property on the search space. The purpose is to speed up the process of identifying the candidates. In phase II, authors have selected the high utility itemsets from the high transaction weighted utilization itemsets by scanning the database only once. Li *et al.* [8]-[10] developed some efficient approaches, namely, FSM, SuFSM, ShFSM, and DCG methods for share mining. Share mining can be treated as a special class of utility mining. Share mining is equivalent to utility mining under suitable adjustment on item count and external utility of items. Atallah *et al.* [11], proposed a technique that is based on data perturbation *i.e*. they have modified data by a selective set of 1-value to 0- value, in order to lower the support of sensitive rules in such a way so that the utility of the released database is maintained at some maximum value. The efficiency measured is depicted in terms of the number of hidden non-sensitive rules. The process counts the rules that are hidden because of the data modification process. Dasseni *et al.* [12] extended the sanitization of sensitive large itemsets to the sanitization of sensitive rules. They addressed the hiding issue through modified support and confidence. The work either hides the frequent itemsets from which they are derived to prevent the sensitive rules from being generated , or decreases the confidence of the sensitive rules below a user-specified threshold for this purpose. This complete work can be found in [13]. In [14] Stanley *et al.* have developed a framework which is an enhancement of their previous work. The authors here claim to achieve equality between privacy and disclosure of information by attempting to minimize the influence on sanitized transactions. Oliveira *et al.* [15] proposed a heuristic based framework for privacy in data mining frequent itemsets. They gave the algorithm that hide the set of frequent pattern containing the sensitive information. The algorithms rely on an item restricted methodology in order to evade adding of noise to the data. Evaluations of algorithm have been done on the basis of the

parameters like hiding failure, miss cost, and performance. Hiding failure refers to the percentage of restrictive pattern that is discovered, miss cost is the percentage of non restrictive patterns that are hidden after sanitization process and the performance of the algorithm is measured by computation time of sanitization process by keeping constant the size of database and set of restrictive pattern. In Sweeney *et al.* [16] proposed a heuristic-based approach for protecting raw data through generalization and suppression techniques. The method proposed, provide K-Anonymity for hiding sensitive information. Weng *et al.* [17] proposed an efficient algorithm FHSAR for fast hiding sensitive association rules. They scan database only once to minimize the execution time of the algorithm and hide the sensitive information. In this heuristic, first step is to create the prior weight function for each transaction ID so that order of transaction modification is decided. The second step analyses the correlation between SAR and transaction, hence, it is easily selected to modify the appropriate item. Yeh *et al.* [18] address the algorithm for privacy preserving utility mining. They have proposed two algorithms hiding high utility item first (HHUIF) and maximum sensitive itmsets conflict first (MSICF). They hide the high utility items by reducing the utility value below the user specified threshold to achieve privacy preservation. The advantage of user specific threshold is that user can keep the stability between the privacy and disclosure of information. They have compared both the algorithms on the basis of the hiding failure, miss cost and database difference ratio. Xu *et al.* [19] presented the problem utility based anonymization. They have proposed the framework to specify utility of attributes, covering numeric and categorical data and have also developed local recording technique for utility based anonymization which is a bottom up method. It does not split the domain instead it searches only tuples. The major cost of this method is due to bottom up method that uses two level loops for searching. Then it uses top down method in which set of tuples are partitioned in subsets and they are partitioned again into smaller groups which reduces the weighted certainty. Finally they merge the group to meet the k-anonymity requirement. These two methods were compared with nondarian multidimensional method [20]. They measure the anonymization quality using certainty penalty, discernibility penalty and query answering error rate. One more data modification approach which has been used for association rule confusion is data blocking [21]. certain values of data items are replaced with an uncertain value in this method. This replacement of a real value by an unknown value rather than placing a false value is what is required in certain specific applications (*i.e.*, medical applications). An approach which applies blocking to the association rule confusion is presented in [22]. Some variations on the definition of the support and confidence of an association rule are levied by this new special value in the dataset. Thus, the minimum support and minimum confidence are changed into a minimum support interval and a minimum confidence interval respectively. The authors assumes that the confidentiality of data is not violated until the support and/or the confidence of a sensitive rule lie below the middle of these two ranges of minimum support interval and minimum confidence interval values. An algorithm used for rule confusion in such a case, both 1-values and 0-values should be mapped to certain value in an interleaved fashion. An extension of this work, can be found in [23] with an elaborated discussion on how effective this method can be on restructuring the confused rules. In [24] authors have used the data distortion technique in which the position of the sensitive item is changed without changing its support value. The size of the database remains intact. The approach first uses representative rules to prune the association rules and then hides only those rules that are sensitive. Advantage of this approach is that it is able to hide maximum number of rules however; algorithm fails to hide all the rules, which are otherwise supposed to be hidden in minimum number of passes.

3. Proposed Algorithm

The proposed algorithm is based on decreasing the utility values of each sensitive itemsets by modifying their quantity value so that they are hidden and do not come in the category of high utility items. To facilitate the understanding of the algorithm we first present the notations that are used along the paper and are described in Table 1:

Table 1. Notations with Description

Symbol	Description
I	A set of n distinct item $I = \{i_1, i_2, i_3, \ldots . i_n\}$
TD	A set of m different transaction $TD = \{TD_1, TD_2, TD_3 \ldots . TD_m\}$
$ltu\,(i_p, TD_q)$	Local transaction utility value *i.e.*, the numeric value of item i_p in the transactionTD_q . For example: In Table 2(a)$ltu\,(A, TD_8)$ = 4 and $ltu\,(C, TD_2)$ = 6 *etc.*.
$etu\,(i_p)$	External transaction utility value of the corresponding itemi_p, For example: In Table 2(b)$etu\,(B)$ = 150 and $etu\,(C)$ = 10 *etc.*.
$u(i_p, TD_q)$	Utility value of item that can be calculated through $ltu(i_p, TD_q) * etu\,(i_p)$
$l(S, TD_q)$	Utility value of itemset$S$ in transaction TD_qcan be calculated as$\sum_{i_p \in I} u(i_p, TD_q)$
$l(S)$	Utility value of itemset I in all transaction can be calculated as $\sum_{\substack{TD_q \in TD \\ S \subseteq TD_q}} l(S, TD_q)$
ε	Minimum utility threshold

Utility mining is to find all the itemsets whose utility values are beyond a user specified threshold. An itemset$S$ is a high utility itemset if $l(s) \geq \varepsilon$, whereε is the minimum utility threshold, otherwise, it is a low utility itemset. For example, in Table 2, $l(\{C, D\}) = l(\{C, D\}, T_1) + l(\{C, D\}, T_3) + l(\{C, D\}, T_5) + l(\{C, D\}, T_6) + l(\{C, D\}, T_9) = 24 + 24 + 31 + 23 + 20 = 122$. If $\varepsilon = 85$, $\{C, D\}$ is a high utility itemset. and $l(\{A, C, D\}) = l(\{A, C, D\}, T_1) + l(\{A, C, D\}, T_3) = 27 + 27 = 54$. If $\varepsilon = 85$, $\{A, C, D\}$ is a low utility itemset.

Table 2(a). Transaction Database

TD	A	B	C	D
TD_1	1	0	1	14
TD_2	0	0	6	0
TD_3	1	0	2	4
TD_4	0	0	4	0
TD_5	0	0	3	1
TD_6	0	0	1	13
TD_7	0	0	8	0

TD_8	4	0	0	7
TD_9	0	1	1	10
TD_{10}	0	0	0	18

Table 2(b). Utility Value

Item Name	Profit
A	3
B	150
C	10
D	1

High utility itemsets: Itemset$S$ is a high utilityitemset, if $l(s) \geq \varepsilon$, where ε is the minimum utility threshold.

Sensitive itemsets: Sensitive itemsets is set of subsets of all high utility itemsets for particular minimum utility threshold ε, where each subset is an itemset that should be hidden according to some security policies.

Privacy threshold: As per the [25], privacy threshold is the ratio of sensitive patterns that are still discovered from the sanitized database.

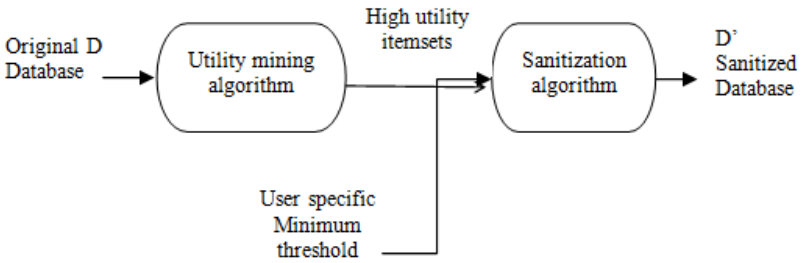

Figure 1. Sanitization Process

Sanitization Process: In sanitization process, first we apply the utility mining algorithm on original database D and generate the high utility itemsets. Then, on the basis of business requirement we find out the sensitive itemsets and input the user specific minimum threshold ε. Finally we apply the sanitization algorithm to hide the sensitive item using minimization of the utility value of the item which is either equal to or below the user specific minimum threshold ε. The sanitize algorithm guarantees that sanitized database will not reveal any sensitive itemsets. The whole process is represented by the Figure 1.

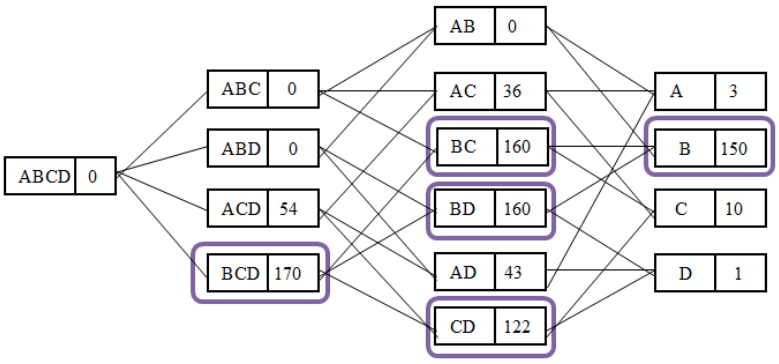

Figure 2. Utility Itemsets with their Respective Utility Value

The working process of this algorithm can be understood with the help of an example: The transactional database Table 2 has been taken from [26] and minimum utility threshold is taken to be 85 *i.e.* $\varepsilon = 85$, as an input. First the algorithm finds all high utility itemsets. The itemsets whose utility value is $\geq \varepsilon$ are: ({BCD}, {BC}, {BD}, {CD} and {B}) and are therefore considered as sensitive itemsets and are represented in Figure 2. The Algorithm selects the itemset {BCD}, where B has highest utility value in transaction {TD9} among all itemsets {BCD}. Now the Algorithm selects item B and modifies its value from 1 to 0 with this the utility value of itemset {BCD} reduces to 20, which is below the minimum utility threshold. Along with the above value the utility value of itemsets {BC}, {BD} and {B} also reaches below minimum utility threshold. Now again the algorithm select next sensitive itemset {CD} which lies in the transaction {TD1, TD3, TD5, TD6, TD9}, where C has the highest utility value in transaction {TD5}. The Algorithm selects item C and modifies its value from 3 to 0. With this the utility value of itemset {CD} reduces to 92 which is not below the minimum utility threshold. Until the utility value of itemsets {CD} does not reaches the minimum utility threshold, the algorithm again selects the item C, where C has next highest utility value in {TD3} and modifies its value from 2 to 0, with this the utility value of itemset {CD} reduced to 72 which is below the minimum utility threshold. The whole working process is depicted by the Figure 3.

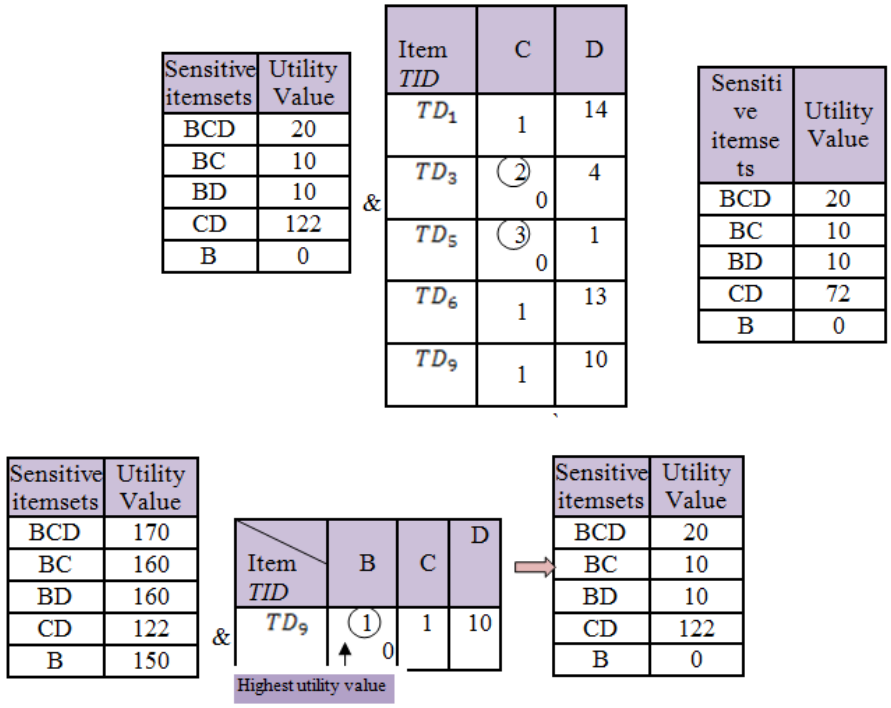

Figure 3. Working Process to Reduce Utility Value of the Sensitive Itemsets

Hiding Algorithm for High Utility Item Iteratively (HAHUII)

Input
(1) A source database D
(2) A min_utility threshold .
(3) A sensitive high utility itemdets.

Output
A sanitized database D' whose utility value of all sensitive itemset below the min_utility threshold.

Step 1 – Input the database D and explore the high utility item sets and their utility values $l(SI_i)$using two phase algorithm.

Step 2 – On the basis of business requirement, Input the minimum utility threshold ε and sensitive high utility item sets.

Step 3 – Utility value of sensitive item require to minimized as $minn = l(SI_i) - \varepsilon$

Step 4 – Select the high utility sensitive item with Max$|SI_i|$. The cardinality of a set *S* is denoted by $|S|$.

Step 5 – Select the highest to lowest utility value item from the sensitive item sets as $(i_p, TD_q) = \max_{i_p \in SI_i}(u(i_p, TD_q))$

Step 6 – Modify the utility value until the value reaches below the minimum utility threshold ε as

$l(SI_i, TD_q) = l(SI_i, TD_q) - \lceil minn/(etu(SI_i)) \rceil$

If the value of $l(SI_i, TD_q)$ is negative then modify utility value by zero.

Step 7 – If the utility value of all the sensitive itemset below the minimum utility then stop the sanitization process. Otherwise, go to step 4 for select next high cardinality sensitive item set and repeat step 5-7.

Step 8 – Finally, obtain the sanitized database D'.

Advantages of the proposed approach is that it uses minimum passes to decrease the utility value of the sensitive itemsets because the algorithm modify utility value iteratively until the value reaches below the minimum utility threshold ε.

4. Results and Discussion

In this section, proposed algorithm is implemented using Java and modifies the sensitive data using Microsoft SQL server. System used for result generation is a Lenovo workstation with core *i*3 Pentium processor, 2 GB main memory and windows7 operating system. The work has used synthetic data containing different number of transaction like 10, 100, 200, 500 and 1000 with 5 distinct items. The proposed approach is compared with previously existing approach [18].To evaluate the effectiveness of the algorithm we measure the algorithm on the following parameters as follows:

Average Execution Time is the average time taken in execution of the algorithm with respect to the number of transactions.

Performance of the algorithm is evaluated on the basis of time which is the time required by the algorithm in sanitization process. Figure 4 shows the performance of the algorithm as compared to HHUIF in terms of average execution time in ms. The graph shows almost parallel average execution time to that of HHUIF till 500 number of transactions, but a drastic improvement is found as the number of transactions increases. Thus, it gives much better results with higher number of transactions which in itself is an achievement.

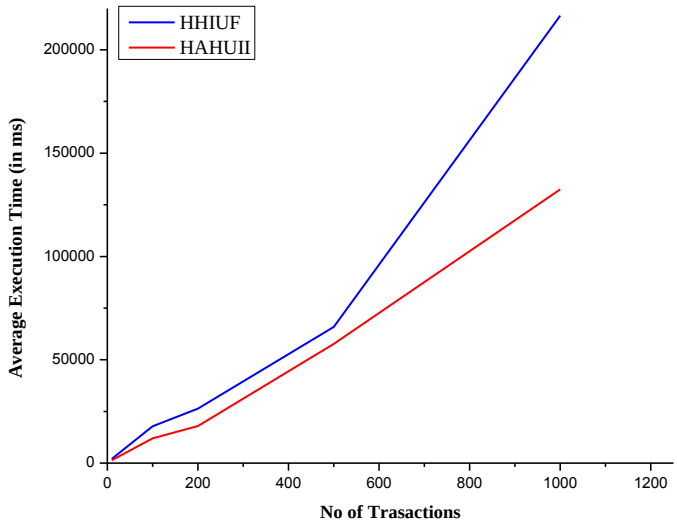

Figure 4. Average Execution Time Comparison for HHIUF and HAHUII

Hiding Failure is the portion of sensitive information which is not hidden by the application of a privacy preservation technique. Another definition says, hiding failure refers to the percentage of restrictive pattern that is discovered. Using the proposed privacy preservation algorithm we have obtained zero hiding failure which signifies that all the patterns that are considered sensitive are hidden well. This hiding failure is calculate using the following formulae:

$$HF = \frac{|\cup(D')|}{|\cup(D)|}$$

Tables 3 (a-c) show the performance of the HAHUII algorithm in terms of hiding failure applied to database with five sensitive itemsets with varying number of transaction from 10, 200 to 500 for several Minutility values and Expected Threshold d. When MinUtility is less than or equal to d, the hiding failure of the algorithm must be 0%. It is possible that when Minutility is greater than d, the value of the hiding failure cab be greater than 0%. In this work, the hiding failure of the algorithm for 5 items with 10 transactions is 100% when the chosen d is quite below the Minutility, hiding failure of the algorithm for 5 items with 200 transactions is 13% when Minutility is equal to threshold and 67% when it is less than threshold. And hiding failure of the algorithm for 5 items with 500 transactions is 21% when Minutility is equal to threshold and 79% when it is less than threshold. Thus, the results depict that hiding failure is completely achieved only when number of transactions are less. There is a deviation from this when the number of transactions is increased and the value of Minutility is equal to threshold. Thus, this is a limitation of the present work.

5. Conclusion and Future Work

Privacy preserving aspect of data mining has gained momentum in the field of research application recently because the data set contains highly sensitive information as well and no user would like that sensitive information should be leaked to outsiders. Therefore, this Sensitive information which can be mined from a database should be excluded, because such sensitive information can equally well compromise data privacy.

The work proposes an innovative algorithm for this purpose. The algorithm can completely hide any given high utility itemsets by scanning the data interactively. This significantly reduces the execution time compared to existing one. Advantage of the proposed approach is that it uses minimum passes to decrease the utility value of the

sensitive itemsets. The algorithm modify utility value iteratively until the value reaches below the minimum utility thresholdε. The proposed approach is compared with previously existing approach as well.

References

[1] P. Tan, M. Steinbach, and V. Kumar, "Introduction to Data Mining," Addison Wesley, **(2005)**.
[2] S. L. Wang, Y. H. Lee, S. Billis and A. Jafari, "Hiding sensitive items in privacy preserving association rule mining", IEEE International Conference on Systems, Man and Cybernetics, vol. 4, **(2004)**, pp. 3239-3244.
[3] V. S. Verykios, A. K. Elmagarmid, B. Elisa, D. Elena, and Y. Saygin, "Association RuleHiding", IEEE Transactions on Knowledge and Data Engineering, vol. 16, no. 4, **(2004)**, pp. 434-447.
[4] E. Dasseni, V. Verykios, A. Elmagarmid and E. Bertino, "Hiding Association Rules by Using Confidence and Support", In Proceedings of 4th Information Hiding Workshop, **(2001)**, pp. 369-383.
[5] S. Shankar, T. P. Purusothoman, S. Jayanthi and N. Babu, "A Fast Algorithm for Mining High Utility Itemsets", In Proceedings of IEEE International Advance Computing Conference, **(2009)**, pp. 1459-1464.
[6] H. Yao, H. Hamilton and L. Geng, "A Unified Framework for Utilty-Based Measures for Mining Itemsets", In Proceeding of the ACM Intel. Conf. on Utility-Based Data Mining Workshop (UBDM), **(2006)**, pp. 28-37.
[7] Y. C. Li, J. S. Yeh and C. C. Chang, "Efficient algorithms for mining share frequent itemsets. In Proceedings of fuzzy logic", In soft computing and computational intelligence 11th world congress of international fuzzy systems association, **(2005)**, pp. 534-539.
[8] Y. C. Li, J. S. Yeh and C. C. Chang, "A fast algorithm for mining share-frequent itemsets", Lecture Notes in Computer Science, vol. 3399, **(2005)**, pp. 417-428.
[9] Y. C. Li, J. S. Yeh and C. C. Chang, "Direct candidates generation: A novel algorithm for discovering complete share-frequent itemsets", Lecture Notes in Artificial Intelligence, vol. 3614, **(2005)**, pp. 551-560.
[10] Y. C. Li, J. S. Yeh and C. C. Chang, "Isolated items discarding strategy for discovering high utility itemsets", Data & Knowledge Engineering, vol. 64, no.1, **(2008)**, pp. 198-217.
[11] M. J. Atallah, E. Bertino, A. K. Elmagarmid, M. Ibrahim and V. S. Verykios, "Disclosure Limitation of Sensitive Rules", In Proceedings of the IEEE Knowledge and Data Engineering Workshop, **(1999)**, pp. 45-52.
[12] E. Dasseni, V. S. Verykios, A. K. Elmagarmid, and E. Bertino, "Hiding Association Rules by using Confidence and Support", In Proceedings of the 4th Information Hiding Workshop, **(2001)**, pp. 369-383.
[13] A. G. Divanis and V. S. Verykios, "Association Rule Hiding for Data Mining", Springer, **(2010)**.
[14] S. R. M. Oliveira and O. R. Zaiane, "Privacy preserving frequent itemset mining", In Proceedings of the IEEE ICDM Workshop on Privacy, Security and Data Mining, **(2002)**, pp.43-54.
[15] S. R. M. Oliveira and O. R. Zaiane, "Privacy Preserving Clustering by Data Transformation", In Proceedings of the 18th Brazilian Symposium on Databases, **(2003)**, pp. 304-318.
[16] L. Sweeney, "Achieving k-anonymity privacy protection using generalization and suppression", International Journal of Uncertainty, Fuzziness and Knowledge Based Systems, vol. 10, no. 5, **(2002)**, pp. 571-588.
[17] C. C. Weng, S. T. Chen and H. C. Lo, "A Novel Algorithm for Completely Hiding Sensitive Association Rules", In Eighth International Conference on Intelligent Systems Design and Applications, **(2008)**, pp. 202-208.
[18] J. S. Yeh and P. C. Hsu, "HHUIF and MSICF: Novel algorithms for privacy preserving utility mining", Expert Systems with Applications, vol. 37, no. 7, **(2010)**, pp. 4779-4786.
[19] J. Xu, W. iWang, J. Pei, X. Wang, B. Shi and A. W. C. Fu, "Utility Based Anonymization for Privacy Preservation with Less Information Loss", UBDM'06, **(2006)**, pp. 440-445.
[20] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan, "Mondrian multidimensional k-anonymity", In Proceedings of the 22nd International Conference on Data Engineering (ICDE'06), **(2006)**, pp. 1-25.
[21] L. W. Chang and I. S. Moskowitz, "An integrated framework for database inference and privacy protection", Data and Applications Security, **(2000)**, pp. 161-172.
[22] Y. Saygin, V. S. Verykios, and A. K. Elmagarmid, "Privacy preserving association rule mining", In Proceedings of the 12th International Workshop on Research Issues in Data Engineering, **(2002)**, pp.151-158.
[23] Y. Saygin, V. Verykios and C. Clifton, "Using unknowns to prevent discovery of association rules", SIGMOD Record 30, no. 4, **(2001)**, pp.45-54.
[24] A. Agrawal, U. Thakar, R. Soni and B. K. Chaurasia,, "Efficiency Enhanced Association Rule Mining Technique", International Conference on Parallel, Distributed Computing technologies and Applications (PDCTA-2011), **(2011)**, pp. 375-384.

[25] A. Chaturvedi, D. N. Goswami, R. Soni and B. K. Chaurasia, "Secure Multi-party Communication in Data-mining Applications", In International Journal of Database Theory and Application, vol. 8, no. 4, **(2015)**, pp. 299-306.
[26] H. Yao, H. J. Hamilton and C. J. Butz, "A Foundational Approach to Mining Itemset Utilities from Databases", In 4th SIAM International Conference on Data Mining, **(2004)**, pp. 482-486.

