www.ijecs.in
International Journal Of Engineering And Computer Science ISSN: 2319-7242
Volume 5 Issues 8 Aug 2016, Page No. 17525-17529

Differentially Private Utility Item Mining

Ms. Chanchal Rathi, Mr. J. Ratnaraj Kumar
Computer
G. S. Moze College of Engineering, Balewadi
Pune, India
chanchal.rathi@gmail.com

Computer
G. S. Moze College of Engineering, Balewadi
Pune, India
ratnaraj.jambi@gmail.com

Abstract— In today's business world there is excess of available data & a great need to make good use of it. Data mining is art of extracting pattern and knowledge from large amount of data. Frequent itemset plays essential role in many Data Mining tasks that try to find out interesting patterns from database such as association rules. Association rule mining is a finding association among large databases variables. Mining of frequent itemset is most popular problem in data mining. The frequent itemsets recognition is valuable for economic & research purpose. But valuable discovered frequent itemsets should not only assure security but also achieve high data utility & offer time efficiency. The frequent itemsets are patterns or items like itemset, substructures or subsequences that comes out in data set frequently. There are several Frequent Item mining algorithms for frequent item mining such as Apriori, Frequent Pattern growth, Eclat, Utility Pattern growth algorithms.

To provide security or privacy here we use differentially private Utility Item mining algorithm using Utility Pattern- growth algorithm. It consists of Preprocessing & mining phase. In preprocessing phase, to enhance utility & privacy advance smart splitting method is proposed to transform database. For given Database preprocessing phase should be performed only once. In mining phase run time estimation & dynamic reduction performed. To cover the information loss by smart splitting, we contrive run time estimation to calculate actual support of itemsets in original database. For privacy we have added noise in the database, we put forward dynamic reduction method to reduce the noise dynamically which guarantees privacy during mining process. In this paper we proposed new algorithm for mining high utility itemsets called as UP growth which consider not only frequency of itemset but also utility associated with the itemset.

Keywords— Frequent Item Mining, ϵ- differential privacy, FP- growth, UP-growth.

I. INTRODUCTION

Data mining is called as uncovering hidden data in a database. In other word, it is called as also data analysis, data driven determination and deductive finding out. Among the areas of data mining, the problem of extracting associations from data has received a great deal of awareness. Association rules are used to identify correlations among a set of items in database. These correlations are based on existence of the data items & their properties.

Market basket analysis is application of Association Rule Mining. The market analysts focused in discovering frequently bought items by customers, so the organization can do effective arrangements of items according to their sales. Two strategically measures that command the association rule mining process are support and confidence. Support is the statistical importance of a rule while confidence is the degree of assuredly of the detective associations the whole association mining process is commanded by two variables, minimum support and confidence which are user defined.

Discovering useful patterns hidden in database plays an important role in different data mining jobs, such as frequent pattern mining, high utility pattern mining. Frequent pattern mining is a research topic that has been used to different databases having long transactions. It is used in the analysis of purchase of customer transactions in retail research where it is called as market basket analysis. It is used to identify the purchase patterns of the consumer. Given a database, where each transaction has a set of items, FIM tries to find itemsets that occur in transactions more often than a given threshold. The frequent itemsets detection can provide, if the data is intuitive (e.g., web browsing history and medical records of patients), releasing the detected frequent itemsets might cause threats to individual privacy.

This paper addresses the frequent and weighted itemsets discovery, i.e., the frequent weighted itemsets, from transactional weighted data sets using UP tree and UP growth algorithm for high transaction itemsets.

II. LITERATURE SURVEY

Lots of studies have been proposed to solve the privacy preserving FIMproblem from different from different aspects. Main aim is to ensure that the resulted frequent itemsets itself does not leak private information and achieve differential privacy. Considering K-anonymity model for protecting privacy in [2], [12] propose an algorithm to publish annonymised frequent itemset. These two studies don't satisfy differential privacy. And thus they cannot provide sufficient privacy protection from attackers having background knowledge. [3] A new novel and powerful privacy definition called l –diversity. [3] Show the weak points of k-anonymity. Diversity framework introduced here to give strong privacy guarantee.

[4] Proposed fast algorithm for mining association rule i.e. Apriori & AprioriHybrid algorithms .These compared with previous algorithms and these algorithm gives excellent performance for large database with transactions, but these generates candidate set.

[5] Introduces FP growth algorithm, with is nothing but mining frequent pattern without candidate generation, as

we have seen in [4] apriori algorithm performs mining fastly with candidate set generation, which is costly. In [5] FP tree is used as data structure to store large database compressed in small data structure. Algorithm introduced in [5] is scalable and efficient than apriori algorithm.

[11] Present set of randomization operators to limit privacy beaches in FIM. [13] Proposed new algorithm for Transaction often than minimum support threshold is subset of some basis with differential privacy guarantee.
But [6] [13] [14] addresses some issues performing frequent item mining with differential privacy.

III. PROPOSED APPROACH WITH FRAMEWORK AND DESIGN

Problem statement:

- The existing system does not find utility transactional itemsets.
- Existing methods require more time for mining.
- Existing system gives no. of output combinations, so it's not accurate.

A. System Architecture

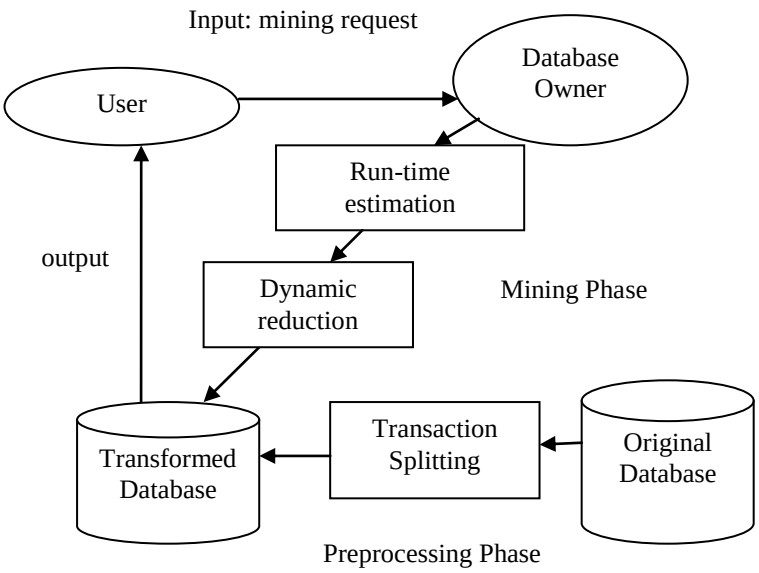


Figure 1. System Architecture

B. Proposed system

Differentially private UIM algorithm based on the UP-growth algorithm & provides differential privacy to protect data.

UP growth algorithm:
It is a partitioning based, depth first search algorithm.
It adopts divide and conquer manner to adopt to decompose the mining task into smaller tasks. To enhance mining performance & avoid database scan again & again, UP-growth uses compact data structure called UP tree. UP growth uses Header Table & UP- tree as data structure.

discovering frequent patterns in sensitive data adopted exponential mechanism & Laplace noise-addition mechanism techniques which are efficient in context of frequent item mining.

[14] Proposes algorithm Privbasis with perform frequent itemset mining with differential privacy by using minimum support threshold. An item set that found in

UP tree have root node and its child nodes. In which each node represents item name & related information to it such as count means support of item, preceding node, successor node, its linking with other node having same name.

Header table consist of item name, calculated utility value & link.
UP-growth algorithm consists of 3 steps

1. Construct UP tree.
2. UP tree generates high utility itemset.
3. From Possible high utility itemset recognize high utility itemset.

UP Growth Algorithm:

1. Get transaction file as input to system Tp= {I, Iq,Ip}
   I= Itemset in transaction
   Iq= Itemset quantity
   Ip= Profit of each itemset in transaction
2. Calculate the transaction utility of each transaction using following formula,

Transaction utility of a transaction Td is denoted as TU (Td)
TU (Td) $=\sum_{k=0}^{n} Iq * Ip$

3. Calculate transaction weighted utility of each item.

Transaction-weighted utility of an itemset X is the sum of the transaction utilities of all the transactions containing X, which is denoted as TWU(X) and defined as

$$= \sum X \subseteq Td \wedge Td \varepsilon D$$

If TWU(X) > minimum support (min_sup)
Remove itemset X from transaction T.
Else construct UP tree

4. UP tree Construction

- Up tree= { N.name, N.count, N.nu, N.predecessor, N.hlink}
- N.name= name of node.
- N.count= support count of node.
- N.nu= Node utility of node.
- N.predecessort= parent of node.
- N.hlink= path traversal of nodeins_tran(N, i_x).

- If N has a child N_{ix} such that N_{ix}.item=i_x, increment N_{ix}.count by 1. Otherwise, create a new child node N_{ix} with N_{ix}.item=i_x, n_{ix}.count=1, N_{ix}.parent= N and N_{ix}.nu=0
- Increase N_{ix}.nu by RTU(t_j)- $\sum_{p=x+1}^{n} u(ip, tj))$, where $i_p \varepsilon t_j$.
- If x≠n, call ins_tran(N_{ix}, i_{x+1})

5. UP-Growth Procedure:

UP-Growth (*T*R, *H*R, X) is called, where *T*R is the UP-Tree and *H*R is the header table and X is itemset.
For each entry *ai* in *H*x do
Generate the phui for each item as follows

- Generate a PHUI $Y = X \cup ai$;

- Set pu(i_k) as estimated utility of Y
- Path utility of item ik in {im}-CPB is denoted

as pu(ik, {im}-CPB) and defined as the following equation:

- If path utility of that item is less than min utility then remove that item from that path. And calculate the new PU.

Put local promising items in Y-CPB into H_Y
Apply DLU to reduce path utilities of the paths
Apply DLN

Algorithm consist of

1. Preprocessing phase
2. Mining phase.

Preprocessing phase:-

Utility and privacy trade-off can be improved by using transaction splitting techniques. To improve privacy utility trade off transactions are splitted rather than truncated. Smart splitting is performed in this phase.
By extracting the information from original database smart splitting is performed and original database is transformed. For given Database preprocessing phase performed only once.
We are introducing dataset used & metrics computed.
Mining phase:-
In this phase, given the noisy database and a user-defined threshold, it privately discovers frequent itemset. To enhance quality result two methods are used i.e. Run time estimation & dynamic reduction.
In this phase we divide privacy budget ϵ in to 5 portions.
$Є_1$ is used to compute maximum length constraint
$Є_2$ is used to estimate maximal length of frequent itemsets.
$Є_3$ is used to reveal correlation between items in transaction.
$Є_4$ is used to compute vectors of itemsets.
$Є_5$ is used to compute support.

C. Algorithms

Algorithm 1: Long transaction splitting

Input: Long transaction t of length p, maximal length constraint L_m, correlation tree
Output:q=(p/ L_m) subsets.

1. Set R as zero initially. Consider leaf nodes those items are in transaction t, remove items from leaf node those are not in t & create initial node N_l.
2. Initially transaction is empty, select node with higest number of items.
3. Add items in n_l to the transaction and remove n_l from $N_{l..}$
4. Sort remaining nodes in $N_{l..}$
5. For each node in N_l do step 6.
6. If transaction & n_l' count is less than or equals to maximal length constraint then add items in n_l' to transaction and remove same items from $N_{l.}$
7. Add transaction into R
8. For each node in N_l randomly add the items in node to subsets in R.
9. Give R

pu(ik,{im}–CPB)= $\sum_{\forall p \subseteq ik \quad p \in \{im\}-CPB} PU(Ik, \{im\} -$ path utility of each path of that item towards the root is that item;s node utility.

- Calculate the path utility of each item included in given paths by using above formula.

Algorithm 2: Preprocessing Phase

Input: Original database D, Percentage n, Privacy budgets $Є_1$, $Є_2$, $Є_3$
Output: noisy database D'
1. α = Get noisy number of transactions with different lengths using $Є_1$.

1. Get maximam length constraint using α & percentage n.
2. β = Get noisy maximal support of transactions of different lengths using $Є_2$.

3. Z=calculate r*n matrix using micro vectors of itemsets.Z will be used in runtime estimation for calculating information loss by transaction splitting.
4. D_1 = Execute length constraint on original database D by random truncating.
5. Set2=with the help of $Є_3$ compute noisy support of all 2-items in D_1
6. Make unidirectional weighted graph G which is Set2 based
7. Create Correlation tree from unidirectional weighted graph & length constraint
8. Initially noisy database is empty. For transactions in database D,
9. If transaction exceeds length constraint then go to step 11 else go to step 12
10. create sub transactions & add each subset in ST with weight 1/ST into D'
11. Add transaction to noisy database
12. Give noisy database.

Algorithm 3: Mining Phase
Input: Transformed database D', Privacy budget $Є_4$, $Є_5$, matrix Z, threshold λ, maximal length constraint L_m, array β
Output: utility iemsets
1. L_f = calculate maximal length of utility itemsets based on β & α.
2. Repeat step 3 L_f times.
3. Get noisy result of Z's i^{th} row by using $Є_4$/ L_f.
4. Set utility items & header table as zero & Є'as Є5/ L_f.
5. Calculate noisy support of an item. For each item in alphabet, add Laplace noise to its support, using noisy support do the runtime estimation method in which calculate average & maximal support in original database.
6. If maximal support exceeds threshold value then do step 7
7. Add item into header table
8. if average support of item exceed threshold value then do step 9
9. Display item as utility item.
10. Construct UP-tree based on maximal supports in header table. In which arrange items in header table in decending order by considering value of their maximal support.
11. Considering header table and UP tree we get thee conditional pattern base data.

DOI: 10.18535/ijecs/v5i8.16

12. By using this data finally we get the utility itemsets.

D. Mathematical Model

System's mathematical model represent as "S"
Where S={Input,Output,Process}
Each of above phase is describe as follow.
Input: {Databases file having long transactions}
Output :{utility itemsets whose support exceed maximal constraint}
Process:
Given the alphabet I = {i1; . . . ; in}, a transaction t is a subset of I and a transaction database D is a multiset of transactions. Each transaction represents an individual's record.

A non-empty set X subset of I is called an itemset, we say a transaction t contains an itemset X if X is a subset of t. The support of itemset X is the number of transactions containing X in the database. An itemset is frequent if its support is no less than the user-specified minimum support threshold.

Let A be a differentially private algorithm for the transformed database and f be a function that can divide one transaction into at most k subsets. Then, for any neighboring databases D and D', and any subset of outputs S subset of Range (A), we have
$Pr(A(f(D)) = S) \leq e^{k-\epsilon} Pr(A(f(D')) = S)$
Where
D&D'-Neighbouring databases
Consider two neighboring databases D and D'. Let t denote the transaction in D' but not in D (i.e. D'= D + t). Suppose the transformed database of D is ~D and t is divided into k subsets t1; ; tk. Since A is a ϵ-differentially private algorithm for the transformed database ~D, based the definition of differential privacy, for any subset of outputs S subset of Range (A), we have.
$Pr(A(\sim D) = S) \leq e^{k-\epsilon} Pr(A(<D', t_1, \ldots, t_k>) = S)$

We can estimate the average support of X in the original database (i.e., ω_a) as

$$\omega_a = \mathbf{avg\_supp(\tilde{\omega},i)} = \int_{\acute{\omega}=\tilde{\omega}-5}^{\acute{\omega}=\tilde{\omega}+5} \mathbf{Pr}(\acute{\omega}|\tilde{\omega})\, a$$

We can estimate the "maximal" support of X in the original database (i.e., ω_m) as

$$\omega_m = \mathbf{max\_supp(\tilde{\omega},i)} = \int_{\acute{\omega}=\tilde{\omega}-5}^{\acute{\omega}=\tilde{\omega}+5} \mathbf{Pr}(\acute{\omega}|\tilde{\omega})\, m$$

Where

- Noisy support of an i-itemset X in the transformed database

-X's actual support in the transformed database
We calculate the Relative error of released itemset supports (RE)

$$\mathbf{RE = median_x\,(|sup_x' - sup_x|/\, sup_x)}$$

Where
X- Generated frequent itemsets
Sup_x -actual support of itemset x
sup_x' – noisy support of itemset x

IV. PRACTICAL RESULTS AND EVIRONMENT

A. Input Dataset :

We use four publicly available real datasets.
Dense datasets: Pumsb-star (PUMSB) and Accidents
Sparse datasets: BMS-POS (POS) and Retail

B. Hardware and Software used :

Hardware configuration

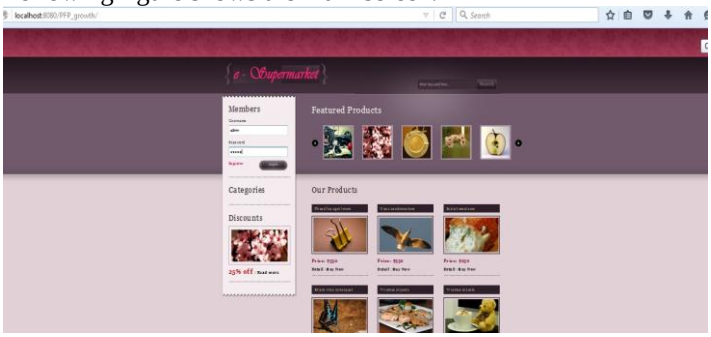
Processor:	Pentium IV
Speed:	2.6 GHz
RAM:	512 MB DD RAM
Hard Disk:	20GB
Keyboard:	Standard Windows Keyboard
Monitor:	SVGA 15'' color

Software configuration

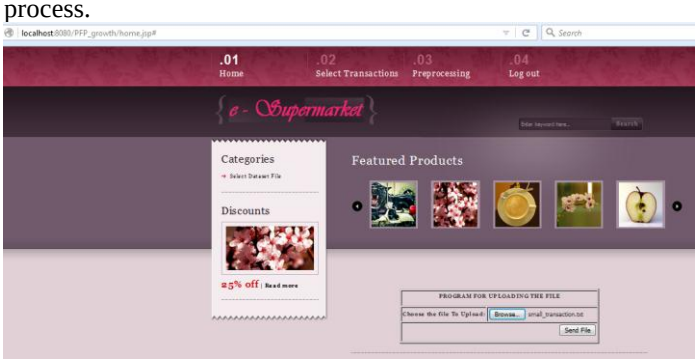
Front end:	Java
Back end:	MYSQL 6
Tools used:	NetBeans IDE 8.0.2
Operating System:	Windows XP/7/8

C. Result of practical work:

Following figures are showing results for practical work done.
Following figure shows the main screen.

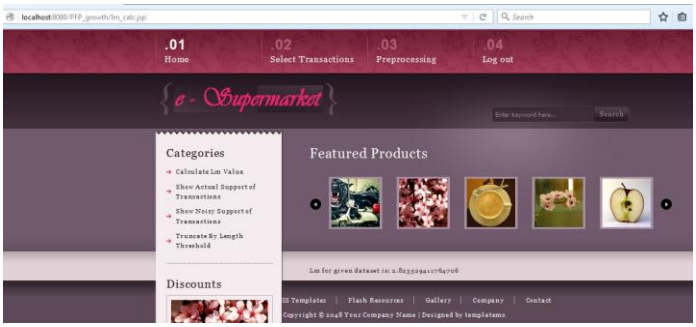


The following window shows that uploading of filefor mining process.



The following window shows different work operations.

The following window shows calculated maximum length constraint value.



Finally conclusion and future work is predicted in section V.

V. CONCLUSION

In this paper we survey some frequent item mining with privacy methods. We studied about the method, which is useful for privacy such as K-anonymity-diversity, Privbasis. We have studied and analyses these methods observed drawbacks and benefits of these methods. We have studied different mining algorithms such as Apriori, Apriorihybrid, FP –growth, UP –growth algorithm performed comparisons between Apriori and FP –growth & UP growth. Apriori is costly to perform and not time efficient than FP –growth algorithm. We studied existing system .We compared the FP-growth with UP-growth algorithm. We conclude that UP growth algorithm is time efficient and requires less memory as compared to FP growth especially when database contains lots of long transactions.

ACKNOWLEDGMENTS

I would like to thanks to my seniors who guided me when ever asked queries to them. Especially thanks to my project guide Mr. Ratnaraj Kumar has given valuable guidance to me. Finally thanks to my family members for giving moral support to me.

REFERENCES

[1] Sen Su, Shengzhi Xu, Xiang Cheng, Zhengyi Li, and Fangchun Yang, "Differentially private frequent item mining via transaction splitting",2015,In IEEE transactions on knowledge and data engineering vol. 27, No.7, pp.1875-1891

[2] C.Dwork, "Differential privacy," in Proc. Int. Colloquium Automata, Languages Programm., 2006, pp. 1–12

[3] L. Sweeney, "k-anonymity: A model for protecting privacy," Int. J.Uncertainity Fuzziness Knowl.-Base Syst., vol. 10, no. 5, pp. 557–570,2002.

[4]] A. Machanavajjhala, J. Gehrke, D. Kifer, and M.Venkitasubramaniam, "l-diversity: Privacy beyond k-anonymity, "in Proc. 22nd Int. Conf. Data Eng., 2006, p. 24.

[5] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," in Proc. 20th Int. Conf. Very Large dataBases,1994,pp.487–499.

[6] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," in Proc. ACM SIGMOD Int. Conf. Manage.Data, 2000, pp. 1–12.

[7] C. Zeng, J. F. Naughton, and J.-Y. Cai, "On differentially private frequent itemset mining," Proc. VLDB Endowment, vol. 6, no. 1,pp. 25–36, 2012.

[8] J. Vaidya and C. Clifton, "Privacy preserving association rule mining in vertically partitioned data," in Proc. 8th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2002, pp. 639–644.

[9] M. Kantarcioglu and C. Clifton, "Privacy-preserving distributed mining of association rules on horizontally partitioned data,"IEEE Trans. Knowl. Data Eng., vol. 16, no. 9, pp. 1026–1037, Sep.2004.

[10] W. K. Wong, D. W. Cheung, E. Hung, B. Kao, and N. Mamoulis,"Security in outsourcing of association rule mining," in Proc. 33rd Int. Conf. Very Large Data Bases, 2007, pp. 111–122.

[11] W. K. Wong, D. W. Cheung, E. Hung, B. Kao, and N. Mamoulis,"An audit environment for outsourcing of frequent itemset mining,"Proc. VLDB Endowment, vol. 2, no. 1, pp. 1162–1173, 2009.

[12] A. Evfimievski, R. Srikant, R. Agrawal, and J. Gehrke, "Privacy preserving mining of association rules," in Proc. 8th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2002, pp. 217–228.

[13] Maurizio Atzori, F. Bonchi, F. Giannotti, and D. Pedreschi,"Anonymity preserving pattern discovery," VLDB J., vol. 17,no. 4, pp. 703–727, 2008.

[14] R. Bhaskar, S. Laxman, A. Smith, and A. Thakurta, "Discovering frequent patterns in sensitive data," in Proc. 16th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2010, pp. 503–512.

[15] N. Li, W. Qardaji, D. Su, and J. Cao, "Privbasis: Frequent itemset mining with differential privacy," Proc. VLDB Endowment, vol. 5, no. 11, pp. 1340–1351, 2012.

[16] F. McSherry and K. Talwar, "Mechanism design via differential privacy," in Proc. 48th Annu. IEEE Symp. Found. Comput. Sci., 2007,pp. 94–103.

[17] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in Proc. 3rd Conf. Theory Cryptography, 2006, pp. 265–284.

[18] Vincent S. Tseng, Cheng Wei wu, Bai- En shei, Philip S. Yu"UP-Growth: An Efficient Algorithm for High Utility Itemset Mining", Department of Computer Science, University of Illinois at Chicago, Chicago, Illinois, USA

[19] Frequent itemset mining dataset repository [Online]. Available: http:// fimi.ua.ac.be/data, 2004.