A Review Paper on MAC Encrypted Frequent Pattern Mining with Uncertain Data

[1] Ankur V.Vadhel, [2] Ravi K.Sheth
[1] Student M.Tech(Cyber Security), [2] Assistant Professor (IT)
[1,2]Department of Information Technology
[1,2]Raksha Shakti University, Gujarat-Ahmedabad, India.

ABSTRACT

There are miscellaneous current working algorithms provides mining in Unstructured/Frequent patterns from existing or precise data based on its importance. Meanwhile now in the era of technology, the need of Uncertain Data Mining is highly increased. There are too many cases in which obtained data are unstructured where shorting is required. For Frequent/Unstructured Data Mining through pattern mining, there are two mainly approaches are providing where one is step-by-step approach and another one is pattern-growth approach. One of the best example of Step-By-Step algorithm is U-Apriori and UF-growth & UFP-growth algorithm are the perfect illustration of Pattern growth approach. While in mining process the data is manipulated differently where security of data in highly needed. There are plenty of chances of data loss in the mining process. There we can provide MAC security to the algorithm that checks the data coming for the mining is form as single MAC address. So we can prevent the collision of different data in Mining process. In the real world the final outputted data is also not as much secured as it is in manipulation process so it is a huge need of security after the Mining where we can encrypt Patterned/Structured data with the help of any encryption key like Ciphertext. In this paper we are carrying out a case study of different existing world algorithm that are working on mining unstructured pattern from Uncertain data to regulate the data obtained from existing world applications with MAC and encryption technologies.

***Keywords**: Frequent Pattern Mining, MAC Protection, Unstructured Data, Data Mining Algorithms, Anonymization, Encryption, Minimum Support, Frequent Patterns, Tree Structures, And Uncertain Data*

I. INTRODUCTION

As an essential work of data mining task, frequent pattern mining focuses to discover implicit, pasty undiscovered and required useful knowledge revealing patterns on collections of consequently or co-occurring items or situation that are uncounted in data. Now in current era of technology, frequent pattern mining is widely used in different existing world business, country, and Rocket-science applications (e.g., banking, biometric, educational modeling, finance, marketing, medical diagnosis, meteorological data analysis). Unstructured/Uncertain data are present in very of these applications. Uncertainty can be caused by (a) our limited perception or understanding of current situation; (b) Cons of the observation equipment; or (c) lack of available objects that need for the counting, storage, transformation, or calculating of data. It can also be out come in nature. Data collected by chemical, electromagnetic, mechanical, optical radiation, thermal sensors [12] in environment surveillance, security, and production systems can be environment hardly. Dynamic errors—such as (i) outcome measurement inaccuracies, (ii) frequency of the sensors, (iii) deviation occurred by a frequently change of the calculated property over time, (d) wireless communication errors, or (e) network fault also introduces uncertainty into the data encountered by these sensors.

There are too many key models and algorithms have been created over the last few years for various Unstructured/uncertain data mining tasks. These include (a) clustering uncertain data, (b) classifying uncertain data and (c) outliers from uncertain data. We examine another data mining task namely, uncertain frequent pattern mining. To mine frequent patterns from uncertain data, different theory that can be applicable. Out of this all, possibility of theory is more famous and largely used by development and research work.

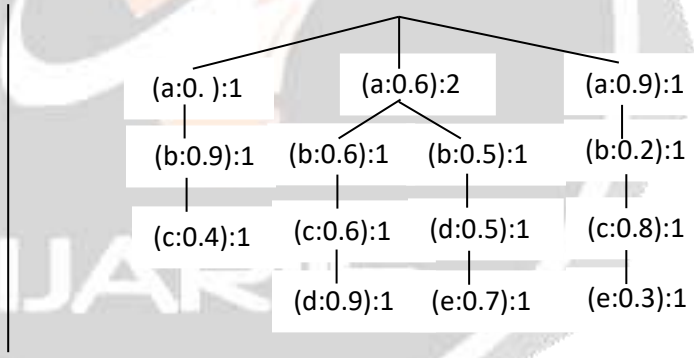
II. TREE BASED ALGORITHMS

I. UF-growth

The patterns frequency of mine from estimated datasets of indefinite data[2], defined a tree-based mining algorithm called UF-growth. Alike to its opposite part for mining exact data (the FP-growth algorithm), UF-growth also builds a tree structure to cover the data of the datasets. However, it doesn't use the FP-tree (as in FP-growth) because every node in the FP-tree only maintains (i) one thing and (ii) its presence count in the tree path. When mining exact data, the real support of an pattern X relays on the happening counts of items inside X. However, when mining indefinite data, the expected support X of X is the addition of the multiplication of the occurrence count and living probability of each thing inside X. Hence, every node in the UF-tree contains of three parts: (i) an item, (ii) its living probability, and (iii) its presence count in the way. Such a UF-tree is built in a alike fashion as the build in the FP-tree, expected that new transfer is joined with a baby node only if the alike thing and the alike living possibilities exist in two transaction and the baby node. As such, it may take to a down compression ratio than the actual FP-tree. Luckily, To decrease the memory usage, UF-growth incorporates two well improved techniques.

`

Item	Cap
a	2.3
b	2.2
C	1.8
d	1.4
E	1.0



(The UF-tree for the probabilistic dataset D2 of uncertain data)

Each frequent sets. One by one frequent patterns for one level to other levels. (By covering the tree route and reduction the occurrence counts) in the mining Process. One hand as paths in a UF- tree are show if there are in same item and same existential probability, The UF-Tree just give only if they have same item and same existential Probability, The UF-Tree meticulously capture the content (Existential Probabilities) Datasets of uncertain data which give results of mined without producing false positives or false negatives.

II. UFP-growth

By Reducing tree size, the tree compact get reduced (reduction in the number of tree nodes) in **the UFP-Growth Algorithm.** As UF-growth and UFP-growth algorithm also give the probabilistic datasets of uncertain data and developed a **UFP-Tree**. As leaf for item x having same existential probability the well of dependent clustered into a mega node. The decrease mega node in the UFP-Tree captures (i) item sets x, (ii) The highest range of probability value (among all nodes within the cluster), (iii) eventuality count. Tree paths are split if the roots on these paths give the same item but similar existential probability values of paths. In other words, we can say that the root sharing condition is less restrictive of the UF-tree. Same time the approximate nature (e.g. The maximum existential probability amount of among all the roots clustered into a mega-node) of UFP-growth also give some infrequent pattern itemsets, (i.e. some false positives) in that frequent pattern itemsets (i.e. true positives). As the third scan of the probabilistic dataset of uncertain data in the give the required these false positives.

III. LITRACURE REVIEW

A. PUF- Tree: A Compact Tree Structure for Frequent Pattern Mining of Uncertain data

As this Pattern, Leungand Tanbeer roll his eyes over that (1) the transaction cap gives CUF-growth including a tighten upper bound to expected support of pattern, (2) in that an the upper bound can be strengthen in tree- based data mining structure. They launch the theory of a prefixed item cap, can be described as bellows. [2]

As per the theorem, the prefixed item cap can be represented through $I^{cap}(X_r,t_i)$ of an item X_r in a process $t_i = \{X_1,X_2,.....,X_n\}$, where $1<r<h$(i.e., $h=|t_i|$), which is described as a outcome of $P(X_r,t_i)$ and the top most required probability of values of M items through X_1 to $X_{(r-1)}$ in t_i (i.e., in the proper prefix of X_r in t_i).

As Perfectly defined as

$$P\ I^{cap}(x_r,t_i) = P(x_r,t_i) \times M, \text{ if } |t_i| > 1$$

$$P(x_1,t_i), \text{ if } |t_i| = 1 \text{(i.e., } t_i=\{x_1\})$$

Where $M = \max q_{\in[1,r-1]} P(x_q, t_i)$.

We can consider that items are grouped in the format of _a, b, c, d, e_ through the bottom to top. Then table presents the prefixed item cap for each item in a process in a predefine dataset D2 of unstructured data. As the figure describes that how there prefixed items are placed in a TREE formation which defined as PUF-Tree, and the same as the algorithm is defined as **PUF-growth** which mines unstructured/uncertain frequent dataset patterns like as UFP-growth, as the same algorithms also provides 3 checks of the dataset of uncertain data to structure the frequent pattern. In very first check, PUF-Growth counts the prefixed item caps where in next check, PUF-Growth constructs a PUF-Tree to encounters (1) an item and (2) its corresponding prefixed item cap. [2]

As per those in CUF-Tree, rots in the PUF-Tree are splits if the nodes on these root splits the same item. So the out coming PUF-Tree is of the exact size as the CUF-Tree, which can be as extracts as the FP-Tree. The top table considered with the PUF-Tree provides the required support of frequent 1-itemsets. The prefixed item caps in the PUF-Tree gives tighten upper bounds to the needed support of k-itemsets where every $k \geq 2$and for any of the k-itemset X, where if the upper bound to its resulting support is less than $min_{support}$, then X can be safely pruned.

Item	Cap
a	3.4
b	3.3
C	2.9
d	2.5
E	2.1

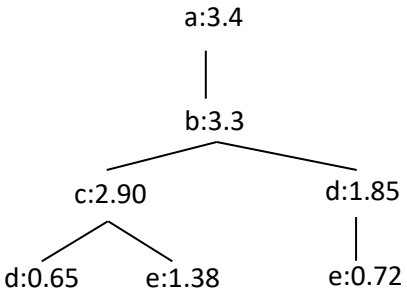


Fig: 1 The PUF-Tree for the dataset D2 of Structured/Uncertain data) [2]

PUF-growth then randomly checks the dataset a 3rd times to scan every of them to verify whether or not they are actually structured (i.e., prune false positives). As represented Table gives, the prefixed item caps which tighten the upper bound to the expected support of unstructured patterns. As same, the number of false positives that need to be consider by PUF-Growth in process of the 3rd check of the dataset [2] of Unstructured/Uncertain data is regularly tiny than that by CUF-growth. Thus, PUF-growth executes quickly in compare to CUF-growth.

B. On **the Security of Two MAC Algorithms:**

In Message Authentication Code (MAC) algorithms is symmetric key which give Data Origin Authentication/data integrity. Message Authenticator Algorithms (MAA) is an ISO standard. There are some technique in which using a confidential key is part of the input to an unkey function. The methods is by Tsudik consist of respectively and confidential secret keys K_1 and K_2 to input. $MAC(x) = h(k_1 \|x\| k_2)$. Internet proposed standard suggest by the IP security (IPSEC) working in group of Authentication of IP Diagram. There are many variant using MD5 function in MAC. Its Allows a variable lengths key, and its support for bitlengths up to 128 bits. The security of message authentication algorithms considered MD5 based method. Eg A 128 bit key can be recovered using 2^{67} known text MAC pairs and time plus 2^{13} chosen text. MAC will check which ip address does the user is login and checking the keys the same hash function has been used when algorithms when is instead in to a system.

C. DISC: Efficient Uncertain Frequent Pattern Mining with Tightened Upped Bounds:

In Uncertain data mining UF- Growth is a tree based algorithms for mining frequent. Its calculates the predict support of a pattern, its essential a significant amount of storage space to collect all existential probability of different values among in the items. To extinguish the more space request of UF-Growth, CUF-Growth algorithm gather nodes having the same item by storing an upper bound on calculate support. In this paper we (i) Its new scenario of domain items specific capping (DISC) (ii) Three new scalable data analytics algorithms that is used scenario to achieve a tighter upper bounds of CUF-Growth. Results shows the potency of uncertain frequent pattern mining with tightening upper bounds then CUF-growth. The concepts of domain item- specific capping (DISC) for tightening upper bounds on itemsets support, DISC-tree capturing the all data of uncertain databases and algorithms

In tighten upper bound, We present the concept of domain item specific capping (DISC). As its name say DISC involves having caps specific to domain item y_i in a given transaction t_j = {y1------yh }. The idea is that instead the multiplying the biggest existential probability value M1 in t_j by second highest probability value M2 in t_j. DISC multiplies the probability value $P(y_i, t_j)$ of domain item y_i by M1 in t_j. In different words let (i) $M1(t_j)$ = maxq $P(y_q, t_j)$ be the greatest existential probability values among all h items in t_j and (ii) y_g =arg maxq $P(y_q\ t_j)$ be the item

having M1 (t_j) so that M1 (T_j) =P(y_g,t_j). Then tree based mining. Its build the tree structure such that each tree node represents some transactions in uncertain itemsets.

D. Secure Frequent Pattern Mining by Fully Homomorphic Encryption with Ciphertext Packing:

In this locality big data is increasing from outside externalization both data storage. Such Outsourcing is convenient for Users and their security and privacy. Private information can have acquired disloyally by data snooping and control monitoring. In data mining classified into 3 approximate: i) Protecting input privacy, ii) protecting output privacy and iii) Cryptosystem. In every system there are advantage and disadvantage. The computational costs of any input and output has its privacy cannot be assuring. Mining result may not become more ambiguous when any input and output privacy are being used. Cryptosystems require more computational time, both secure computation and mining accuracy.

Our volunteer in threefold. (i) The best of our Knowledge in this implementation of our frequent pattern mining built with the FHE packing methods. (ii) This Algorithms will optimize to pack the components columns-wise in the itemsets matrix data that reduce the number of cipher text and associated tasks. In this paper related works on data mining with cryptosystem which works most on the related on frequent pattern mining. Works on data mining with cryptosystem is classified into two categories (i) multi-party computation (ii) homomorphic encryption. In this proposed some algorithms for pattern mining that aim distributed database while preserving privacy by MPC. The Mohammed etal[15] proposed a secure comparison technique with FHE in the case of two party association in rule of mining due to in the storage, communication and computational[15].

E. BLIMP: A Compact Tree Structure for Uncertain Frequent Pattern Mining:

The relative algorithm mines frequent patterns from this compact tree structure. It shows compactness of our tree structure and the tightness of upper bounds to expected assist provided by our uncertain frequent pattern mining algorithm[5].

A roots-level item prefixed-cap tree (BLIMP-tree) which can be as compact as the original FP tree and PUF-tree and a mining algorithm (namely, BLIMP-growth), which finds all frequent patterns from uncertain data.

To tighten the upper bound for all k-item sets (k>2), we propose a branch-level item prefixed-cap tree structure (BLIMP-tree)[5] . The key is to keep track of a value calculated the maximum of all existential probabilities for the single item represented by that node. Every different time a frequent extension is added to the suffix item to form a k-itemset (where k>2), this "blimp" value will be used. Hence, each node in a BLIMP-tree contains: (i) an item xr, (ii) an item cap ICap(xr,t j) and (iii) a blimp value, which is the maximum existential probability of xr in tj (c) shows the contents of a BLIMP-tree for the database. With this information, BLIMP-trees give a tightened upper bound on the expected support of an itemset by the product of ICap (xr,t j) and the "BLIMP" values in the prefix of xr. This new compounded item cap of any k-itemset X = {x1, x2,...,xk} in a tree path tj = x1,x2,...,xh(denoted as I(X,tj) where xk = xr) can be defined as follows. Let tj = x1,x2,...,xr,...,xhbe a path in a BLIMP-tree, where h = |tj| and r ∈ [1,h].

To developed a BLIMP-tree, we scan the transactional database of uncertain data to compute the expected support of every domain item. If Any frequent items are removed. Then we scan the database a second time to insert each transaction into the BLIMP-tree. An item is inserted into the BLIMP-tree according to a predefined order. If a node carrying that item already exists in the tree path, we (i) update its item cap by adding the current ICap (xr,t j) with the existing item cap value and (ii) update its "blimp" value by

taking the maximum of the current P(xr,t j) with the existing "blimp" value. Otherwise, we create a new node with ICap(xr,t j) and P(xr,t j). For a better understanding of BLIMP-tree construction [5].

Sr no	Algorihtm	Advantage	Disadvantage
1	U-Apriori Algorithm	This algorithm greatly reduces the size of candidate set.	It scans database many times and thus performance is affected.
2	UF-growth Algorithm	This algorithm uses UF-trees to mine frequent patterns from uncertain databases in two database scans	It contain a distinct tree path for each distinct items, Existential probability pair
3	UFP growth Algorithm	This algorithm scans the database twice, As nodes for item having similar existential probability values are clustered into a mega node, the resulting mega-node in the UFP-tree.	It contains a distinct tree path for each distinct item, existential probability pair
4	PUF growth Algorithm	This algorithm Mines frequent pattern with constructing a projected database for each potential frequent pattern and recursively mine its potential frequent extensions.	It creates some false positives.
5	BLIMP Algorithm	It takes minimum database scan.	It Create false positives.

IV. CONCLUSION

Throughout the research work, extreme study of the consequence pattern mining algorithms of Unstructured/uncertain item data is calculated and discovered too many advantages and disadvantages of each and another. The new coming current existing algorithms are tested with classical mining algorithms and turns in significant positive and negative points. As another aspect is security as the data is very top classified and the risk of data hacking is highly increased in data mining process so there are many certain ways in which we can secure the data by covering it by MAC address and encrypt the data with any encryption key. This comparison may also take place into various optimization issues that will progress to better performance. Efficiency of the mining algorithms is a need to develop methods to get excellent results.

V. REFERENCES

1. Carson Kai-Sang Leung "Uncertain Frequent Pattern Mining" C. C. Aggarwal, J. Han (eds.), Frequent Pattern Mining Springer International Publishing Switzerland 2014
2. Leung, C.K.-S.and S.K.Tanbeer. "PUF-Tree: A Compact tree structure for frequent pattern mining of uncertain data." In: J. Pei, T., Suzuki, E., Ting, K.M., Inokuchi, A. (eds.) PAKDD 2013. LNCS (LNAI), vol. 7818, pp. 13–25. Springer, Heidelberg (2013).
3. Kai-Sang Leung, Mark Anthony F. Mateo, Dale A. Brajczuk- "A Tree-Based Approach for Frequent Pattern Mining from Uncertain Data", Springer-Verlag Berlin Heidelberg 2008.
4. Radhika Ramesh Naik, Prof. J.R.Mankar- "Mining Frequent Itemsets from Uncertain Databases using probabilistic support", International Journal -2013.

5. Carson Kai-Sang Leung and Richard Kyle MacKinnon- "BLIMP: A Compact Tree Structure for Uncertain Frequent Pattern Mining." Springer International Publishing Switzerland 2014.
6. Carson Kai-Sang Leung, Richard Kyle MacKinnon, Syed K. Tanbeer- "Fast Algorithms for Frequent ItemSet mining from Uncertain Data", IEEE International Conference-2014
7. Tan, P.-N., Chawla, S., Ho, C.K., Bailey, J. (eds.) PAKDD 2012, Part II. LNCS (LNAI), vol. 7302, pp. 322–334. Springer, Heidelberg (2012)
8. L. Wang, R. Cheng, S. D. Lee, et.al, "Accelerating probabilistic frequent itemset mining: a model-based approach, " In CIKM'10, Toronto, Ontario, Canada, pp.429–438, 2010.
9. Toon Calders, Calin Garbini, Bart Goethals, "Approximation of Frequentness Probability of Itemsets in Uncertain Data, " in ICDM, pp.749-754, 2010
10. Calders, T., Garboni, C., Goethals, B.: Approximation of frequentness probability of itemsets in uncertain data. In: IEEE ICDM 2010, pp. 749–754 (2010) .
11. Calders, T., Garboni, C., Goethals, B.: Efficient pattern mining of uncertain data with sampling. In: Zaki, M.J., Yu, J.X., Ravindran, B., Pudi, V. (eds.) PAKDD 2010, Part I. LNCS (LNAI), vol. 6118, pp. 480–487. Springer, Heidelberg (2010).
12. Jiang, F., Leung, C.K.-S.: Stream mining of frequent patterns from delayed batches of uncertain data. In: Bellatreche, L., Mohania, M.K. (eds.) DaWaK 2013. LNCS, vol. 8057, pp. 209–221. Springer, Heidelberg (2013)
13. B. Dong, R. Liu, and W. H. Wang, "Integrity verification of outsourced frequent itemset mining with deterministic guarantee," in Proc. IEEE ICDM 2013, pp. 1025–1030.
14. H. Liu, P. LePendu, R. Jin, D. Dou, "A hypergraph-based method for discov- ering semantically associated itemsets. ," in Proc. IEEE ICDM 2011, pp. 398– 406.
15. J. Liu, K. Wang, B. C. M. Fung, "Direct discovery of high utility itemsets without candidate generation," in Proc. IEEE ICDM 2012, pp. 984–989.
16. H. Liu, P. LePendu, R. Jin, D. Dou, "A hypergraph-based method for discov- ering semantically associated itemsets. ," in Proc. IEEE ICDM 2011, pp. 398– 406. [15] J. Liu, K. Wang, B. C. M. Fung, "Direct discovery of high utility itemsets without candidate generation," in Proc. IEEE ICDM 2012, pp. 984–989.
17. C. K.-S. Leung, "Uncertain frequent pattern mining," in Frequent pattern mining, pp. 417-453, Oct. 2014.
18. [C. K.-S. Leung, P. P. Irani, and C. L. Carmichael, "WiFIsViz: effective visualization of frequent itemsets," in Proc. IEEE ICDM 2008, pp. 875–880.
19. C. K.-S. Leung and Q. I. Khan, "DSTree: a tree structure for the mining of frequent sets from data streams," in Proc. IEEE ICDM 2006, pp. 928–932.
20. C. K.-S. Leung, R. K. MacKinnon, and F. Jiang, "Distributed uncertain data mining for frequent patterns satisfying anti-monotonic constraints," in Proc. IEEE AINA Workshops 2014, pp. 1–6.
21. C. K.-S. Leung, M. A. F. Mateo, and D. A. Brajczuk, "A tree-based approach for frequent pattern mining from uncertain data," in Proc. PAKDD 2008, pp. 653– 661.