IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 6, June 2016

A Proficient Algorithm to ensure Differential Privacy in Frequent Item-set Mining

Akanksha Bhalerao-Kulkarni[1], Prof. Soumitra Das[2]

Research Scholar, Computer Engineering Department, Dr. D.Y. Patil School of Engineering, Pune, India [1]

Head of Department (Computer Engineering), Dr. D.Y. Patil School of Engineering, Pune, India [2]

Abstract: The mining of successive examples is a central part in numerous information mining undertakings. A lot of research on this issue has prompted the excessive need of efficient and scalable algorithms for mining frequent patterns. Meanwhile, discharging these examples is posturing worries on the protection personal data of the clients participating. In this proposition, we examine the mining of successive examples in a protection saving setting. We propose an approach for differential private frequent item-set mining based on LCM algorithm; we refer it as P-LCM algorithm. P-LCM is extended version on PFP growth algorithm which basically works in two phases as pre-processing and mining phase. The first phase being the pre-processing phase it needs to be performed only once and smart transaction splitting method is used in this phase for improving utility as well as privacy trade off. Second phase limits the information loss caused by splitting as well as reduces the amount of noise added during mining process. LCM is an algorithm which finds all frequent item sets in polynomial time per item set. The closed item-sets obtained earlier are not stored in memory. The computational experiments on real world and synthetic databases exhibit the fact that in comparison to the performance of previous algorithms, our algorithms are faster and also maintain high degree of privacy, high utility and high time efficiency simultaneously.

Keywords: Differential Privacy, Frequent Item-set Mining, Transaction Splitting.

I. INTRODUCTION

Recently, the increasing ability to collect personal data from various users marks a threat to privacy and prevents users from participating in public survey forums. In this paper, we focus on privacy issues that arise of finding frequent item-sets in “transactional” data.

Frequent item-set mining is widely used in many applications, perhaps the best known of which is market basket analysis. The goal of frequent item-set mining is to find sets of items that are frequently bought together, and establish an association rule in them. This influences various business decisions. Huge research has been done on frequent item-set mining by our community.

However, with the exception of the recent work in, a differentially private approach to frequent item-set mining has received little attention. A frequent item-set mining algorithm takes as input a dataset consisting of the transactions by a group of individuals, and produces as output the frequent item-sets.

This immediately creates a privacy concern — how can we be confident that publishing the frequent item-sets in the dataset does not reveal any private information about the participating individuals?

This problem is compounded by the fact that we may not even know what data the individuals would like to protect nor what background information might be possessed by an adversary. These compounding factors are exactly the ones addressed by differential privacy [2].

II. LITERATURE SURVEY

A. Related Work

Many different algorithms have been proposed for frequent item-set mining. From that Apriori and FP-growth are the two most well-known ones.

- **Apriori:**

In particular, Apriori algorithm works as breadth-first search, along with candidate set generation-and-test algorithm. This algorithm would need only single database scan if the maximal length of frequent item-sets is one. Thus with the increase in number of frequent item-sets will promote increase in the number of scans as well. [1].
FP-growth algorithm whereas works as depth-first search algorithm, and does not require candidate generation. As compared to Apriori, FP-growth only performs two database scans, which makes FP-growth faster than Apriori in all cases.

- **FP-Growth**:

The promising features of FP-growth motivate us to design a differentially private FIM algorithm based on it. In this paper, we argue that a practical differentially private FIM algorithm should not only achieve high data utility and a high degree of privacy, but also offer high time efficiency. Although a few differentially private FIM calculations have been proposed, we are not aware of any existing studies that can fulfill every one of these necessities at the same time. It not only achieves the degree of privacy, but also offers high time efficiency. There are some limitations of these existing FIM algorithms such as FP- growth scans only two times hence

DOI 10.17148/IJARCCE.2016.5672

we cannot use long transaction for further mining process which may contains frequent item-sets.[5][6]

B.MOTIVATION

Frequent item-sets are a vital part in numerous information/data mining tasks as they help to find important patterns from databases, such as association rules, correlations etc. Mining of association rules is one of the most popular problems from one of them. Thus supermarket transaction data came as the motivation to searching of association rules that help to check the behaviour of purchased item for future use. Association rules describe how often items are purchased together. Mr. Rakesh Agrawal [6] [9] introduced affiliation rules for finding regularities between items in large-scale transaction data recorded by point-of-sale (POS) frameworks in stores. For example, the summary found in the sales data of a supermarket would indicate that a customer purchasing a washing machine is likely to purchase a microwave oven as well. Such information can be used as the premise for choices about advertising exercises, promotional pricing or product placements. Also Data Mining being the current elective course the basic concepts helped in better understanding of the domain.

C.EXISTING ARCHITECTURE

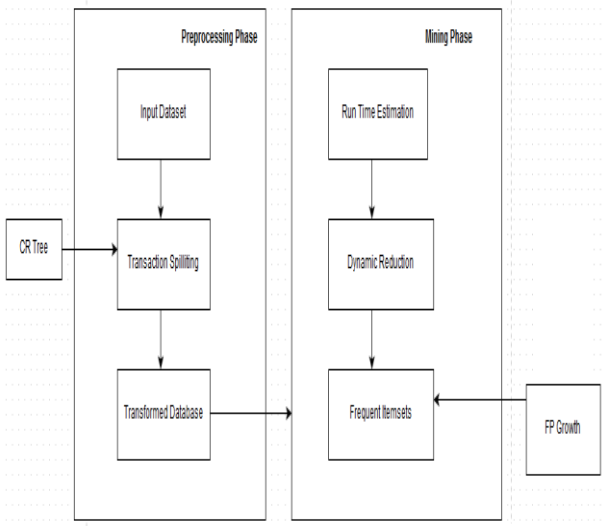


Fig. 1. Existing system architecture with both phases [This diagram is pictorial representation as studied by authors]

III.TAXONOMY CHART

The taxonomy chart denotes the comparison of various existing tools thus giving clarity on constraints and requirement parameters to be worked upon.

Parameters/ References	Tree structure	Efficiency	Time complexity	Performance
Frequent Pattern Growth (FP-Growth) Algorithm	YES	Poor	Low	Average
UP-Growth: an efficient algorithm for high utility itemset mining	YES	Average	High	Average
Mining Frequent Item-sets – Apriori Algorithm	NO	High	High	Average
Differentially Private Frequent	YES	High	Low	Good

IV.PROPOSED FRAMEWORK AND DESIGN

A. PROBLEM DEFINITION

To develop a secure and more accurate system which provides mining strategy of Frequent Item-sets using P-LCM algorithms, this will ensure the reduction in data loss with optimum output.

B. MATHEMATICAL MODEL

Let S be the system which we use to find the private frequent item-sets. FP growth performs well in case of differential privacy for frequent item-set finding.
It consists of two phases:

1. Pre-processing
2. Mining phase

Mathematically it is as follows:
S = {P, M, FIM} where,
P = Pre-processing phase
M = Mining phase.
FIM = Frequent Item-sets.

Input: A transactional data set T= {t1, t2, t3,..., tn} is a set of transactions, where each transaction tq (q belongs to [1,n]) is a set of items in I and each is characterized by a transaction ID (tid) where,
I= {i1, i2,..., im} be a set of data items.

1. **PRE-PROCESSING PHASE**:
Assume that P= {D, N, ϵ1, ϵ2, ϵ3, TS}
Where, D= original database;
N= percentage,
ϵ1, ϵ2, ϵ3 are the privacy budgets,
TS = transaction splitting criteria.

For calculating privacy budgets we need following:

i. Sensitivity[1]:
Given p count queries Q, for any neighbouring databases D;
D’ the sensitivity of Q is:
ΔQ = max ||Q (D) - Q (D’)||.

DOI 10.17148/IJARCCE.2016.5672

The Laplace distribution with magnitude M, i.e., Lap (M), follows the probability density function as
$\Pr[x|M] = 1/2\, M{*}e^{-|x|/M}$,
where $M = \Delta Q / \epsilon$
is determined by both the sensitivity ΔQ and the privacy budget ϵ.

ii. Threshold calculation[1]:
G(ϵ/Cn*Lf)
Where,
ϵ = privacy budget,
Cn is the length of transaction and
Lf is maximum transaction length.

iii. Smart splitting using Weighted Splitting Operation[1]:
Consider a transaction t whose length exceeds the maximal length constraint Lm.

A function f divides t into multiple subsets t1, ..., tk, where ti is assigned a weight wi and the length of ti is under the length constraint Lm.
Then, function f is said to be a weighted splitting operation iff:

$$U^{k}_{i=1}\ ti \text{ and } \sum^{k}_{i=1} (wi \leq 1).$$

Given a transaction t of length p (p > Lm), we aim to partition the p items into q =[p=Lm] subsets t1, ..., tq, each of which satisfies the length constraint, so as to minimize the within subset sum of shortest path lengths:

$$\text{avg min} \sum^{q}_{i=1} \sum^{Iu,Iv\,\epsilon\, ti}_{i=1} \text{dist (iu, iv)}$$

2. MINING PHASE:

MI = {TD, T, PB, Z}
Where,
TD = transformed database,
T = threshold value,
PB = Privacy budget,
And Z = matrix.

Following are the processes from mining phase:
1. Estimate the actual support of transformed database.
2. Estimate the actual support of Original database

Output: FIM (frequently mined item-sets):
We have to perform algorithms i.e. Mining Phase algorithm for frequent item-set mining.

MI= {D, Lm, Lp, Dp, prefix, M, ϵ', upArray}

Where, D = the transformed dataset,
Lm = maximal length constraint,
LP = List,
DP= conditional pattern base,
Prefix= the prefix item-set,
Є' and M are the Privacy budget and threshold respectively, upArray is Up-Array.

Final Output: Frequent item-set F
Where F = {f1, f2,…,fn}

C. SOFTWARE ARCHITECTURE
The new architecture helps us understand how the proposed system functions. The FP-Growth algorithm is replaced with LCM algorithm [9] for better results.

LCM stands for **L**inear time **C**losed item set **M**iner. Already existing algorithms list the final output of frequent item sets with cutting off unnecessary item sets by pruning. However, if pruning is not complete, they continue to function on unnecessary frequent item sets and may ultimately lead to data loss.

In LCM, a parent-child relationship amongst frequent closed item sets is worked upon. This relationship induces tree-shaped transversal routes which consist of all the frequent closed item sets only. Our algorithm traverses the routes taking linear time of the number of frequent closed item sets. LCM is designed on the basis of reverse search technique. LCM-freq by far has significant results as far the prior algorithms are concerned and this is depicted by the results obtained by computer experiments on real datasets.[4][9]

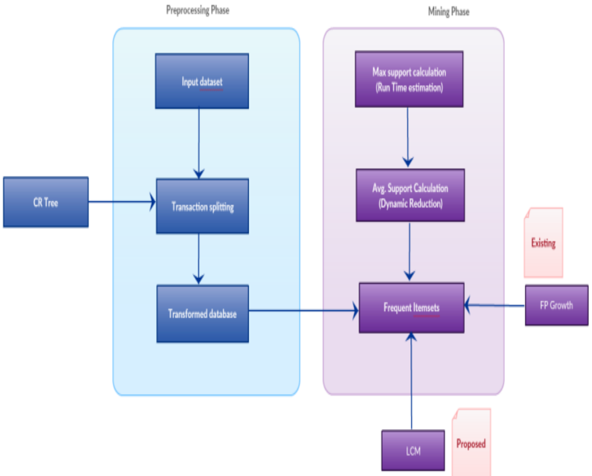


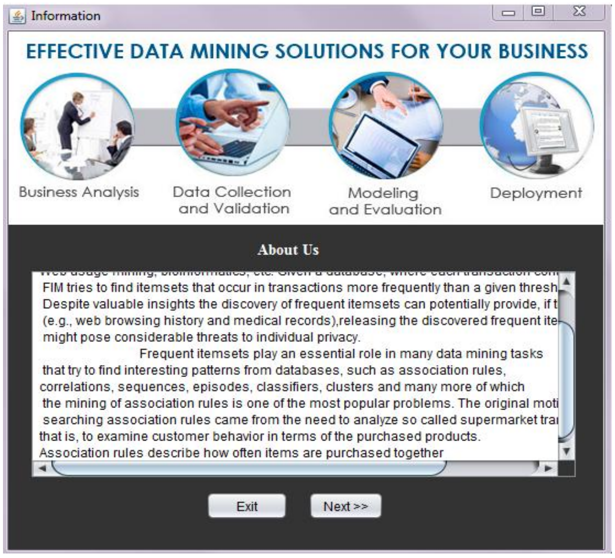
Fig. 2. The System Architecture using P-LCM Algorithm [This diagram is pictorial representation of proposed system designed by authors.]

V. PROJECT MODULES

Module 1: In this module we create Basic GUI of user side. User can insert input transaction dataset through this GUI and pass it for pre-processing steps.

- In this module user or we can say it as **admin**, who can browse input transaction dataset file and upload it for pre-processing operations.
- Then system will do pre-processing operations given in second algorithm of pre-processing. Such as assign privacy budgets, calculate **maximum threshold value** for transaction splitting, create CR tree etc.
- We will create different set of item-sets whose length is greater than calculated maximum threshold value.
- Then we will split long transactions for further mining phase.

DOI 10.17148/IJARCCE.2016.5672



```
Web usage mining, bioinformatics, etc. Given a database, where each transaction con
 FIM tries to find itemsets that occur in transactions more frequently than a given thresh
 Despite valuable insights the discovery of frequent itemsets can potentially provide, if t
 (e.g., web browsing history and medical records),releasing the discovered frequent ite
 might pose considerable threats to individual privacy.
                    Frequent itemsets play an essential role in many data mining tasks
 that try to find interesting patterns from databases, such as association rules,
correlations, sequences, episodes, classifiers, clusters and many more of which
 the mining of association rules is one of the most popular problems. The original moti
 searching association rules came from the need to analyze so called supermarket tra
that is, to examine customer behavior in terms of the purchased products.
Association rules describe how often items are purchased together
```

Fig. 3. The Welcome Screen

Login User
EFFECTIVE DATA MINING SOLUTIONS FOR YOUR BUSINESS
Business Analysis
Data Collection and Validation
Modeling and Evaluation
Deployment
Submit
Reset

Username: admin

Password: *******

Fig.4. The Login Screen

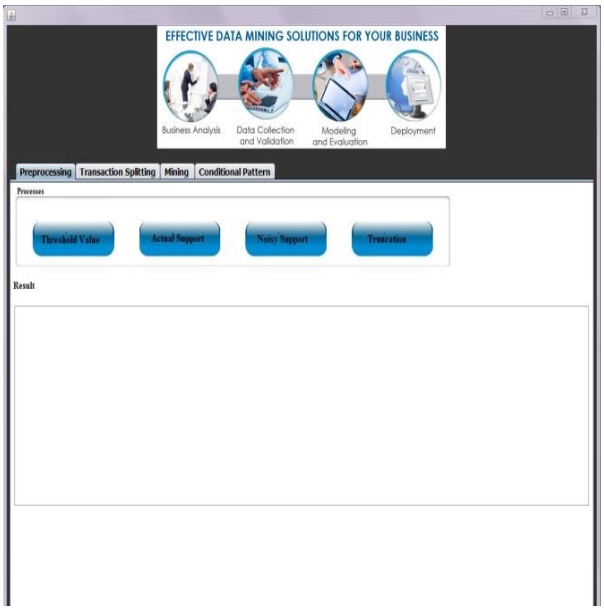
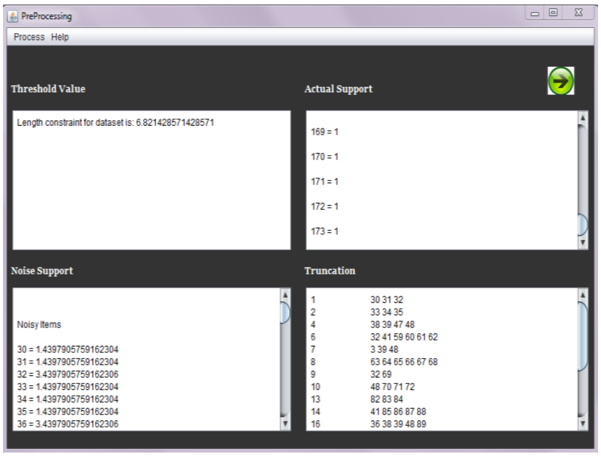


Fig.5. The Home Screen



```
Length constraint for dataset is: 6.821428571428571
```

```
169 = 1
170 = 1
171 = 1
172 = 1
173 = 1
```

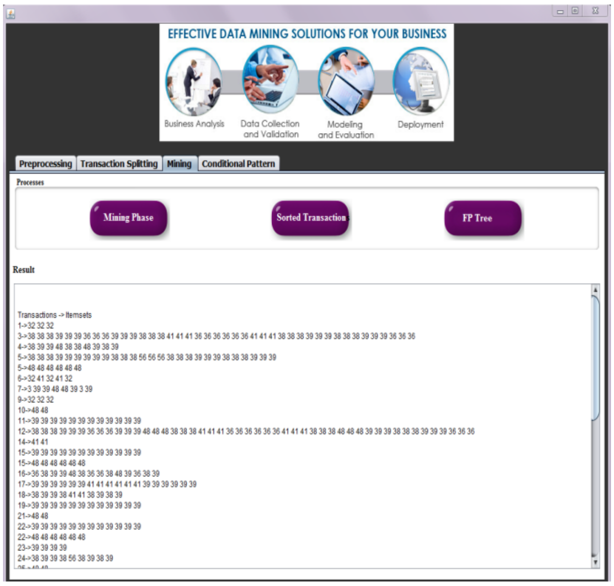
```
Noisy Items

30 = 1.4397905759162304
31 = 1.4397905759162304
32 = 3.4397905759162306
33 = 1.4397905759162304
34 = 1.4397905759162304
35 = 1.4397905759162304
36 = 3.4397905759162306
```

```
1        30 31 32
2        33 34 35
4        38 39 47 48
6        32 41 59 60 61 62
7        3 39 48
8        63 64 65 66 67 68
9        32 69
10       48 70 71 72
13       82 83 84
14       41 85 86 87 88
16       36 38 39 48 89
```

Fig.6. The Pre-processing Phase

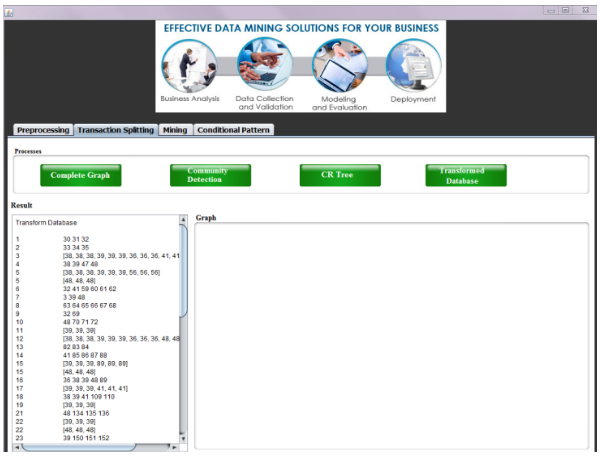
Module 2: This module deals with implementation of Existing system.

- In this module we will implement mining phase using FP-growth algorithm.
- And generate and store its results.



```
Transactions -> Itemsets
1->32 32 32
3->38 38 38 39 39 39 36 36 36 39 39 39 38 38 38 41 41 41 36 36 36 36 36 36 41 41 41 38 38 38 39 39 39 38 38 38 39 39 39 36 36 36
4->38 39 39 48 38 38 48 39 38 39
5->38 38 38 39 39 39 39 39 39 38 38 38 56 56 56 38 38 38 39 39 39 38 38 38 39 39 39
5->48 48 48 48 48 48
6->32 41 32 41 32
7->3 39 39 48 48 39 3 39
9->32 32 32
10->48 48
11->39 39 39 39 39 39 39 39 39 39 39 39
12->38 38 38 39 39 39 36 36 36 39 39 39 48 48 48 38 38 38 41 41 41 36 36 36 36 36 36 41 41 41 38 38 38 48 48 48 39 39 39 38 38 38 39 39 39 36 36 36
14->41 41
15->39 39 39 39 39 39 39 39 39 39 39 39
15->48 48 48 48 48 48
16->36 38 39 39 48 38 36 36 38 48 39 36 38 39
17->39 39 39 39 39 39 41 41 41 41 41 41 39 39 39 39 39 39
18->38 39 39 38 41 41 38 39 38 39
19->39 39 39 39 39 39 39 39 39 39 39 39
21->48 48
22->39 39 39 39 39 39 39 39 39 39 39 39
22->48 48 48 48 48 48
23->39 39 39 39
24->38 39 39 38 56 38 39 38 39
25->48 48
```

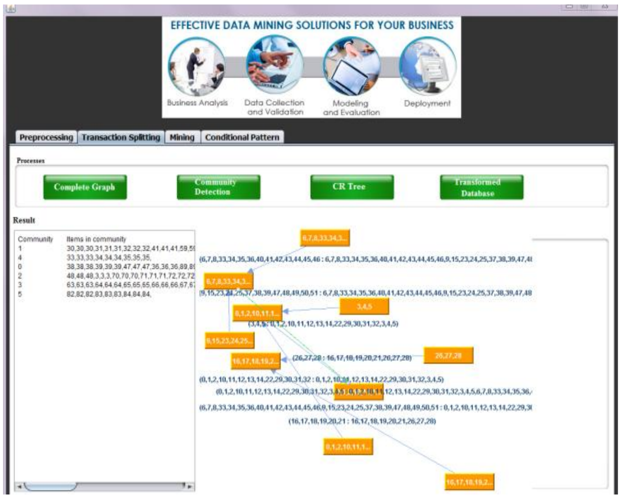
Fig. 7.The Mining Phase



```
Transform Database

1        30 31 32
2        33 34 35
3        [38, 38, 38, 39, 39, 39, 36, 36, 36, 41, 41
4        38 39 47 48
5        [38, 38, 38, 39, 39, 39, 56, 56, 56]
5        [48, 48, 48]
6        32 41 59 60 61 62
7        3 39 48
8        63 64 65 66 67 68
9        32 69
10       48 70 71 72
11       [39, 39, 39]
12       [38, 38, 38, 39, 39, 39, 36, 36, 36, 48, 48
13       82 83 84
14       41 85 86 87 88
15       [39, 39, 39, 89, 89, 89]
15       [48, 48, 48]
16       36 38 39 48 89
17       [39, 39, 39, 41, 41, 41]
18       38 39 41 109 110
19       [39, 39, 39]
21       48 134 135 136
22       [39, 39, 39]
22       [48, 48, 48]
23       39 150 151 152
```

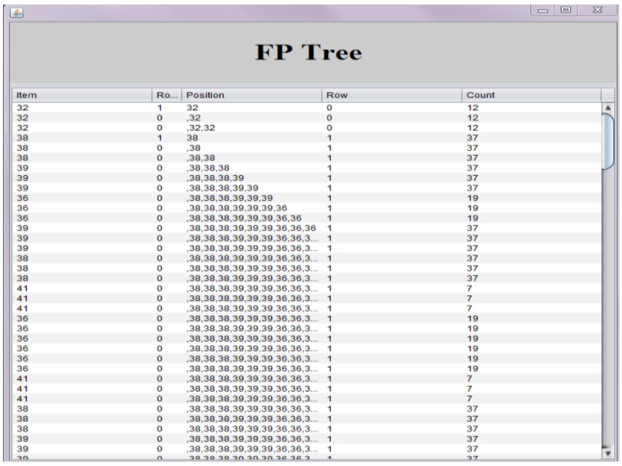
Fig.8.The Transaction-Splitting Algorithm

DOI 10.17148/IJARCCE.2016.5672



Community	Items in community
1	30,30,30,31,31,31,32,32,32,41,41,41,59,5
4	33,33,33,34,34,34,35,35,35,
0	38,38,38,39,39,39,47,47,47,36,36,36,89,8
2	48,48,48,3,3,3,70,70,70,71,71,71,72,72,72
3	63,63,63,64,64,64,65,65,65,66,66,66,67,6
5	82,82,82,83,83,83,84,84,84,

Fig.9.The Complete graph generation


FP Tree

Item	Ro...	Position	Row	Count
32	1	32	0	12
32	0	,32	0	12
32	0	,32,32	0	12
38	1	38	1	37
38	0	,38	1	37
38	0	,38,38	1	37
39	0	,38,38,38	1	37
39	0	,38,38,38,39	1	37
39	0	,38,38,38,39,39	1	37
36	0	,38,38,38,39,39,39	1	19
36	0	,38,38,38,39,39,39,36	1	19
36	0	,38,38,38,39,39,39,36,36	1	19
39	0	,38,38,38,39,39,39,36,36,36	1	37
39	0	,38,38,38,39,39,39,36,36,3...	1	37
39	0	,38,38,38,39,39,39,36,36,3...	1	37
38	0	,38,38,38,39,39,39,36,36,3...	1	37
38	0	,38,38,38,39,39,39,36,36,3...	1	37
38	0	,38,38,38,39,39,39,36,36,3...	1	37
41	0	,38,38,38,39,39,39,36,36,3...	1	7
41	0	,38,38,38,39,39,39,36,36,3...	1	7
41	0	,38,38,38,39,39,39,36,36,3...	1	7
36	0	,38,38,38,39,39,39,36,36,3...	1	19
36	0	,38,38,38,39,39,39,36,36,3...	1	19
36	0	,38,38,38,39,39,39,36,36,3...	1	19
36	0	,38,38,38,39,39,39,36,36,3...	1	19
36	0	,38,38,38,39,39,39,36,36,3...	1	19
36	0	,38,38,38,39,39,39,36,36,3...	1	19
41	0	,38,38,38,39,39,39,36,36,3...	1	7
41	0	,38,38,38,39,39,39,36,36,3...	1	7
41	0	,38,38,38,39,39,39,36,36,3...	1	7
38	0	,38,38,38,39,39,39,36,36,3...	1	37
38	0	,38,38,38,39,39,39,36,36,3...	1	37
38	0	,38,38,38,39,39,39,36,36,3...	1	37
39	0	,38,38,38,39,39,39,36,36,3...	1	37
39	0	,38,38,38,39,39,39,36,36,3...	1	37
39	0	,38,38,38,39,39,39,36,36,3...	1	37

Fig.10. The FP- Tree

Module 3: In this module we experiment our proposed system and compare it with existing system for analysis.

- In this module our proposed P-LCM algorithm is replaced instead of FP growth for mining frequent closed item-sets.
- We plan to integrate LCM algorithm in existing mining phase algorithms and obtain results.
- The results obtained are stored for study of the comparative result of existing and proposed system.

Module 4: In this module we test the new system for expected results.

- In this module the analysis of obtained results is conducted with regards to expected ones.
- Thus statistics is drawn over system performance.

VI. RESULTS AND DISCUSSION

The project is tested for Time efficiency and F-score % and the following results were obtained and plotted. The Existing PFP growth and proposed P-LCM both are compared for two datasets. The Retail and Accident Datasets were used as inputs.

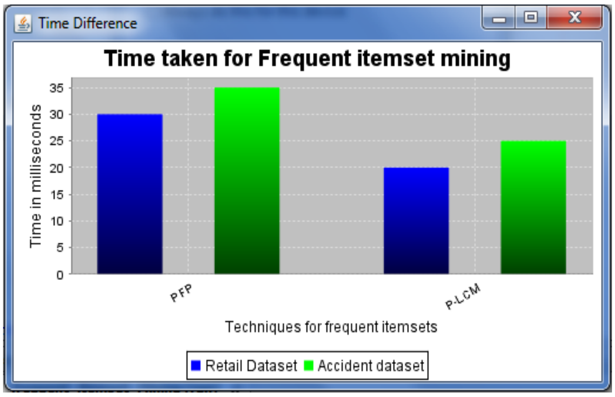


Fig11: Computation time

The above graph clearly indicates that when we used retail dataset as input PFP- Growth Algorithm took 30miliseconds to provide Frequent Itemsets and for Accident dataset as input 35 miliseconds of time was consumed.

Whereas for the same datasets as input P-LCM gave the output in 20 miliseconds and 25 miliseconds respectively. Hence P-LCM proves to be faster.

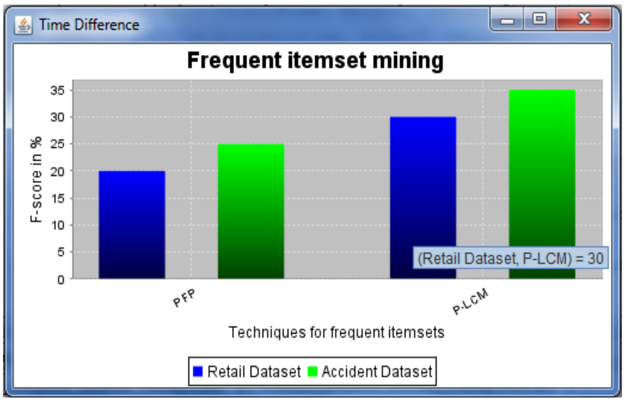


Fig12: F-score %

The F-Score determines the precision and recall value of any algorithm under observation. The higher the score the better. PFP-Growth depicts a score of 20% and 25 % for Retail and Accident dataset as input.

While, P-LCM exhibits a 30% and 35% for the same. Thus evidently P-LCM outperforms the latter.

VI. CONCLUSION AND FUTURE ENHANCEMENT

The need for designing differentially private data mining algorithms has seen growth as for frequent item-set mining purposes. It is the backbone of Data Mining. The most traditional and not much effective algorithms have been the cause behind this development. Thus through this project we intend to provide better and time saving results of frequent item-set mining along with maintaining the security of long transactional datasets. An effort to considerably replace the traditional FP-growth algorithm with P-LCM algorithm is tested for results. The concept of Differential Privacy, Transaction splitting and Run Time Estimation are studied in depth.

DOI 10.17148/IJARCCE.2016.5672

Our future work extends to apply same techniques on higher dimensional dataset of transactions.

ACKNOWLEDGMENT

It gives me great pleasure and immense satisfaction to present this paper which is the result of unwavering support, expert guidance and focused direction of my guide and HOD, **Prof. Soumitra Das** and my M.E. Staff to whom I express my deep sense of gratitude and humble thanks, for their valuable guidance.

REFERENCES

[1] Sen Su, Shengzhi Xu, Xiang Cheng, Zhengyi Li, and Fangchun Yang ‘Differentially Private Frequent Itemset Mining via Transaction Splitting’, 2015 IEEE Transactions on Knowledge and Data Engineering

[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] Takeaki Uno, Tatsuya Asai, Yuzo Uchida, Hiroki Arimura,‘LCM: An Efficient Algorithm forEnumerating Frequent Closed Item Sets’.

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

[6] R. Agrawal and R. Srikant, “Fast algorithms for mining association rules,” in Proc. 20th Int. Conf. Very Large Data Bases, 1994, pp. 487–499.

[7] 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.

[8] 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.

[9] Takeaki Uno1, Tatsuya Asai2, Yuzo Uchida2, Hiroki Arimura2, “LCM: An Efficient Algorithm for Enumerating Frequent Closed Item Sets”, National Institute of Informatics.

[10] Akanksha Bhalerao-Kulkarni,Prof. Soumitra Das,” A New Energy Efficient Vertical Handover Algorithm In Heterogeneous Networks.” www.mjret.in/M61-2-4-10-2016

[11] Akanksha Bhalerao-Kulkarni, Prof.Soumitra Das,” An Algorithm for Frequent itemset mining to incorporate Differential Privacy and to increase proficiency.” www.ijret.net/

DOI 10.17148/IJARCCE.2016.5672