International Journal of Engineering & Technology, 7 (4) (2018) 3412-3417

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET
doi: 10.14419/ijet.v7i4.14609
Research paper

Top-K high utility item set identification in big data by MUP-growth the evolutionary approach with less time constraints

Tejaswini K. Thorat [1] *, Amol D. Potgantwar [1]

[1] Savitribai Phule Pune University
**Corresponding author E-mail: thoratteju@gmail.com*

Abstract

High utility itemset mining is a eminent data mining technique used for acquiring the itemsets with high utility among the transactional dataset. As it supports various proposed analysis, it is adopted in a distinct domain applications, ranging from network to medical records data. At present there is huge amount of data generation from various sources, different algorithms have been promoted to handle such a data and also used to recognize high utility itemsets. This research, evaluates MUP-Growth (Multithreaded Utility pattern growth) algorithm to address the high utility itemset mining problem in big data domain with minimum amount of time constraints. The information of such a high utility itemset is maintained in tree data structure known as UP-Tree(utility pattern tree). In this paper, we propose a new framework for mining top-k high utility itemset, where k is the desired number of HUIs to be mined. Performance of proposed algorithm is computed on different datasets and compared with previous approach. Experimental evaluation shows that proposed algorithm out performs better in terms of time constraints. Finally, based on the research, it gives forthcoming research direction to expand any application in the region of pattern mining by selecting the proper combination of these technologies.

Keywords: *Big Data; High Utility Itemset Mining; Top-K High Utility Item set Mining; HUI-MUP- Growth..*

1. Introduction

In recent years, increasing the amount of the data has become very challenging for data analysis systems. There is a need of an important set of some techniques that can extract the effective and useful knowledge from the huge data. Its very challenging for the researchers, because the traditional techniques of data mining were not able to handle such kind of complicated data in effective manner. By using such traditional techniques researcher has to face problems like some popular techniques had to be redesigned from the scratch for new platform[2].
Every social sites and digital process produce the big data. Huge data is very hard to process as it contains the information of the records of number of people, which includes everyday tremendous data from various fields (like social networking sites, videos etc). In the IT field big data is a largest term. To analyze such a huge data which may be in structured or in unstructured format, big data analytics is essential for processing of complex and huge datasets.
There are some traditional centralized mining algorithms[2]. But these algorithm are efficient only when all the dataset is loaded in the main memory. As they are not designed for the parallel and distributed environment, they can not deal with the big data.
Data mining with the machine learning is the main area of the research on which big data analytics is depend[2]. In big data analysis, pattern mining from large datasets is a basic problem. There are various pattern mining approaches, such as frequent pattern mining[2], [17], [22], [21], [18], [19], weighted pattern mining [12], [11], association rule mining[1].
Among the various approaches for mining essential information from datasets frequent pattern mining[2] technique has been widely applied on various databases, application domains, such as Healthcare to predict next medication[12], traffic data biological data, in decision support system, mobile environment[30], YAFIM algorithm[20] to discover relationship between medicines, [32], [33] Biomedical big data, -omic and EHR data.

Table 1: Transaction Table and their Transaction Utility (TU)

TID	Transaction	TU
T 1	(A,1) (C,10) (D,1)	17
T 2	(A,2) (C,6) (E,2)(G,5)	27
T 3	(A,2) (B,2) (D,6) (E,2)(F,1)	37
T 4	(B,4) (C,13) (D,3) (E,1)	30
T 5	(B,2) (C,4) (E,1)(G,2)	13
T 6	(A,1) (B,1) (C,1) (D,1)(H,2)	12

Table 2: Profit Value of Each Item in Transaction Table

Item	A	B	C	D	E	F	G
Profit	5	2	1	2	3	5	1

Though, importance of each frequent item is not considered in frequent pattern mining approach. Thus, weighted association rule mining was developed [24]. However, with the weight of items, quantities of that items are not considered in association rule mining. Therefore the requirements of retail stores are not satisfy, who are looking for the itemsets with profit i.e., weight as well as purchase quantity of that item.
To overcome this issue, expert presents high utility itemset mining field[31], [29],which is two phase scanning approach. i) In first phase scan, it calculate the Transaction Utility (TU) and Transaction Weighted Utility (TWU) of that itemsets. ii)In second phase scan, generates the PHUI from constructing global UP-Tree [31]. This approach maintains the quantities of that item along with its

weight. It gives better performance and full fill the requirements of the users. Table 1 and table 2 gives the transaction dataset and profit values for each item to calculate importance/quantity of item.

This paper is formulated as: After introductory part, Second section gives definition of some terms that need to be understand. Then carried out a survey of related work in third section. While fourth section introduces the propose work along with system block diagram. Fifth section represents mathematical representation of system. Experimental evaluation is done in sixth section along with that performance measures are stated in seventh section. Afterword, in conclusion section, it is concluded that propose approach performs better than previous approach related to time constraints.

2. Background

In this section, before going through review there are some terms that needs to be identify in problem of utility mining are defined.

As database having finite set of items, , $I=i_1, i_2, .., i_n$,each item i has profit value $pr(i_p)$ which is maintained in the profit table in database as shown in Table 2. Transaction database, as shown in Table I contains the set of transactions like, $D = T_1, T_2, .., T_n$ and each transaction T_d has a unique identifier, TID (Transaction Identifier). Each transaction is maintained with its quantity such as $q(i_d, T_d)$ i.e., combination of item name i_d and its purchase count in T_d transaction.

Definition 1: *Transaction Utility (TU) of transaction is defined as the summation of item's (profit value)*(quantity of item)*

$$TU(T) = \Sigma\, pr(i_d) * q(i_d, T_d) \quad (1)$$

For example, consider T_1 transaction from Table 1 and profit value from Table 2.

$$TU(T) = \Sigma\, pr(i_d) * q(i_d, T_d) \quad (2)$$

$$= pr(A)*q(A, T_d)+pr(C)*q(C, T_d)+pr(D)*q(D, T_d) \quad (3)$$

$$= (5*1)+(1*10)+(2*1) = 5+10+2 = 17$$

Definition 2: *Transaction Weighted Utility i.e., TWU is defined as the summation of all the transaction utility TU of each transaction in which it is occured i.e.*

$$TWU(i)=\Sigma_{\, i \subset T_d \wedge T_d \in D}\, TU(T_d) \quad (4)$$

For example, from Table 1 consider TWU for item A ,

$$TWU(A)=\Sigma\, TU(T_d) = TU(T_1) + TU(T_2) + TU(T_3) + TU(T_6)$$
$$= 17+27+37+12 = 93$$

Definition 3: *Potential High Utility Itemset PHUI is defined as, the itemset with high utility i.e., it is a set of items which having Transaction Weighted utility $\geq$ min utility threshold value. Such as, A: 93, B: 92, C: 99, D: 96, E: 107 from Table 1 and Table 2.*

3. Review of literature

This section represents the various approaches to cope with the big data mining problem. Before reviewing the various research articles for the proposed system, extensive studies have been carried out for mining frequent patterns in big data[2], [15], [21], [18], [19], [16], [20], [17], [22] and issues among them. It is widely recognized that there are various mining algorithms, [2], [21], [17].

Discussed [3], [5], [6]about the big data challenges, security and privacy risks in big data. Survey on big data domain has been carried out[7], [8], [4], [9] and examines various techniques and challenges of big data. Also, states [10] the usage of big data in several areas like data mining, cloud computing, Banking, [35], [34]-omic healthcare data, EHR data, etc.

There are some issues of frequent pattern mining approach like, sequential pattern mining[12], [11]used for identifying transient relationship between medications and Association Rule Mining[1].To address this issue of sequential pattern [13] proposes incremental algorithm to mine HUS from the Incremental Database. Experimentally, they have proved that the proposed algorithm executes much faster than existing algorithm. It does not need to re-scan data hence memory is also saved.

One of the famous algorithm for association rule mining is Apriori[14], [1]used for mining large dataset. To address the issues in traditional apriori algorithm [15] has proposed top k algorithm which is objective directed. As traditional approach (Apriori) does not hold anti monotone pruning strategy. Here, they have proposed such pruning strategy which allows low utility itemsets to be pruned on anti monotone strategy without requiring a user specified utility and hence provides efficient result. In future work, they can include this scheme for FP tree algorithm.

Afterword, pattern growth mining algorithm were proposed like FP-Growth[16], [22], [14] used to mine frequent itemsets. It having various advanced versions such as ,R- PFP growth algorithm to achieve recursion to provide parallelism[22],IPFP an improved PFP algorithm [19],BPFP [17] which uses simple strategy of grouping, YAFIM [21] in medical application to discover the relationship between medicines.

As per the survey carried out on the frequent pattern mining,[14], [16], [2], [22], [20], [18], [23] FP-Growth performs better than apriori as it discover the frequent itemset without generating candidate itemsets [16], [14] and scans the database only twice. As candidate generation is costly specially for long datasets, [16] proposed Frequent Pattern Tree(FP Tree) approach. This approach is scattered in 3 parts; first data is compressed, then FP tree is constructed without using candidate key and then finally divide and conquer technique is applied. They have concluded that FP tree is faster than Apriori Algorithm.

In the framework of frequent itemset mining pattern, item's importance is not considered. Therefore weighted association rule mining was developed[25], [24] have improved the drawback of traditional association rule by providing weights to each items and has proposed Weighted Association Rule (WAR) concepts. Initially they find frequent items without associating weights to the items and then after getting results, they associate weight to those items and then again find frequent items. Experimental result shows that this paper results better results than traditional method but mining performance was not good. To resolve this issue downward closure property was introduced [26].

Although, importance of items is considered with the help of weighted association rule mining, quantity of item in particular transaction is not taken into consideration. Thus, extensive studies [29], [15], [27], [28], [31], [30], [32] have been addressed this issue.[33] has surveyed 2 major problems regarding HUI. Also, when vertical data format was applied on HUI, it had problem to handle the data having similar transactions hence it failed to reduce the size of data base. To overcome all these problems, they have proposed HUITWU algorithm. This algorithm executes result in less time and less space. Then [29] offers an efficient research method for high utility pattern mining for handling incremental databases, while considering many insertions, deletions, and modifications with the currently available memory size. Here, they have proposed three tree structures to efficiently perform incremental and interactive HUP mining. The initial one is Incremental HUP Lexicographic Tree structure, second one is IHUP Transaction Frequency Tree and the next one is IHUP Transaction-Weighted Utilization Tree which is based on TWU value in descending order.

4. Proposed system

The proposed system architecture consists of three steps:

1) It scans the database twice and construct the global UP-Tree, which computes the transaction unit (TU) and TWU transaction weighted unit,
2) In second step, local UP-Tree is generated by removing unpromising items and
3) Finally, it identifies the high utility itemsets.

The overview of the proposed system architecture is illustrated in fig. This fig.1 shows that system having some datasets to discover the potential high utility itemsets from it. In that first data preprocessing is done, afterword global UP-Tree is constructed on the basis of Transaction Utility (TU) and Transaction Weighted Utility (TWU) of the item in the transaction set of dataset. Then, from that TWU , unpromising item less than min utility i.e.,the item which having TWU less than threshold value(min utility) are get removed from transaction to achieve the set of items which having more profit value, which is usefull for the user. Then, local UP-Tree is constructed by removing local unpromising itemset for each item in transaction for each path.

4.1. Proposed data structure: UP-Tree

To improve the performance of mining technique and to avoid the repeated scanning of database, expert uses a compact tree structure, called UP-Tree structure. It uses pruning technique to remove the unpromising items from the dataset. Maintains the data/information of high utility itemset and transaction dataset in tree format. Elements required for constructing UP-Tree and strategy to construct UP-Tree is given in the following section with example.

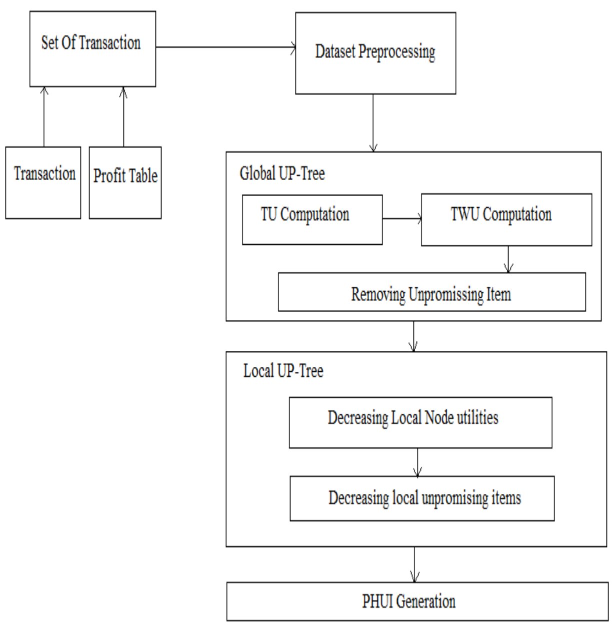

Fig. 1: PHUI Generation Using UP-Tree Data Structure.

4.1.1. Elements of UP-tree

UP-Tree having, N as a element node, N.name gives the name of each node, N.nu is the node utility, N.Parent is the parent node, N.hlink is the node link points to the node having item name same as N.name.

4.1.2. Strategy DGU to construct global up-tree and local up-tree

Proposed system uses two scan approach for mining high utility items from the datasets and it uses tree data structure for the same. After data preprocessing, global UP- Tree is constructed with two scan. In first scan,Transaction Utility (TU) of each transaction and Transaction Weighted Utility (TWU) of each item from the transaction set is computed. Then ,unpromising items i.e., the items whoes TWU is less than the min utility value (threshold) get discarded from the transaction table and again calculate the TU (Transaction Utility) of each remaining transaction by removing the unpromising items called as RTU (Reorganised Transaction Utility).

Definition 4: *Reorganised Transaction Utility RTU is the value obtained from subtracting the profit value of unpromising item with its quantity from Transaction Utility TU of that transaction which contains such unpromising items. i.e.*

$$RTU(T)=TU(T)-pr(upr(i_d))*qupr(i_d,T_d) \quad (5)$$

where, $upr(i_d)$ is unpromising item and $qupr(i_d, T_d)$ is qauntity of unpromising item.

For example ,

From Table 1 TWU of each item is, T W U (A) : 93, T W U (B) : 92, T W U (C) : 99, T W U (D) : 96, T W U (E) : 107, T W U (F) : 37, T W U (G) : 40, T W U (H) : 12. From that, discard global promising items having value less than min utility is 50.Remaining are,A:93,B:92,C:99,D:96,E:107 maintained in header table as shown in Table 3 .Now, from this reorganized items, RTU is computed as shown in Table 4.

Table 3: Itemset after Discarding Unpromising Items with Its RTWU (Reorganized Twu)

Item	E	C	D	A	B
RTWU	107	99	96	93	92

Table 4: Reorganized Transaction Table with their RTU‘s

TID	Transaction	RTU
T 1	(C,10) (D,1) (A,1)	17
T 2	(E,2) (A,2) (C,6)	22
T 3	(E,2) (D,6) (A,2) (B,2)	32
T 4	(E,1) (C,13) (D,3) (B,4)	30
T 5	(E,1) (C,4) (B,2)	11
T 6	(C,1) (D,1) (A,1) (B,1)	10

Then, in second scan construct, global UP-Tree on that transaction database table by using RTU instead of TWU.DGU (discarding global unpromising item) strategy gives us better performance by removing unpromising items from the large datasets. Global UP-Tree is constructed by calculating node utilities N.nu of each node in each transaction as shown in fig. 2.

Definition 5: *Node utility N.nu of each node in the transaction is calculated By, subtracting the summation of each items (profit value) * (quantity of that item) except node in N.nu.*

$$N_{item}.Nu = RTU(T) - \Sigma\, pr(i_d) * q(i_d, T_1) \quad (6)$$

For example, consider transaction T (1) from Table 4.

T (1) = (C, 10) (D, 1) (A, 1)

For first node in transaction where , N_{item} is C ,

$$N_C.Nu = RTU(T_1) - \Sigma\, PR(i_d) * q(i_d, T_1) \quad (7)$$

$= RTU(T_1) - pr(D) *+pr(A) * 1 = 17-(2+5) = 10.$

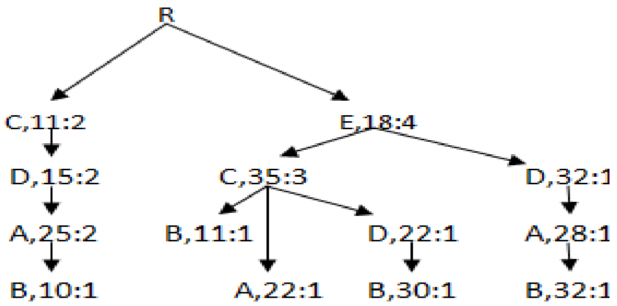

Fig. 2: UP-Tree by Applying Strategy DGU.

Thus ,node utility of node C is 10. Now, for second node D,

$N_D.Nu = RTU(T1) - \Sigma\, pr(i_d) * q(i_{d,}\, T_1)$.

$= RTU(T1) - pr(A) * 1 = 17 - 5 = 12$.

Node utility of remaining item A is,

$N_A.Nu = RTU(T1) = 17$.

4.2. The proposed mining method: MUP-growth

After constructing global UP-Tree, construct the Local UP-Tree, where some part of the global Tree get considered. Afterword, local unpromising items from that tree get removed. Finally, PHUI generation is done using MUP-Growth (Multithreaded UP-Growth) algorithm and performance is increased as it requires less time for computing PHUI by providing multithreading concept.
Most popular approach to mine patterns is FP-Growth [16], [2], but it is observed that FP- Growth does not maintain the quantity of the items in the transaction i.e., profit value is not maintained. To address this issue, MUP-Growth algorithm is proposed which is extended version of UP-Growth algorithm with multithreaded approach. Which gives us the PHUI generation in minimum time constraints along with profit value.

4.2.1. DLU strategy (discarding local unpromising items)

In this, calculation of path utility of the items or each node's path utility in the tree is given.

Definition 6: *Path Utility PU of path P for* $(i_n) - CPB$ *is pu(P,* $(i_n) - CPB$*) ,where* N_{in} *in the node utility and P is the retrieved path from* N_{in} *in the UP-Tree*

For example, consider node B, Here
$\{B\}$–CPB is N_B.nu
Bhlink → B i.e., summation of all the node utilities of the node having same name as B.
Thus, Bhlink B = 10 + 11 + 30 + 32 = 83.Now, by tracing all the path from B resultant paths are,

B= {< ADC >: 10, < CE >: 11, < DCE >: 30, < ADE >: 32}

Now, next step is to calculate local utility i.e., summation of all the utility of node in which they occurred.

$\{A\} = 10 + 32 = 42$

$\{D\} = 10 + 30 + 32 = 72$

$\{E\} = 11 + 30 + 32 = 73$

$\{C\} = 10 + 11 + 30 = 51$

$\{B\} = 83$

Afterword ,remove the local unpromising items i.e., the items which having value less than threshold value. In our example, A having the local utility less than 50 ,which is 42.Thus,remove A from that path by subtracting the profit value from that path utility, i.e.,< ADC >:< DC >= 10 − 5 = 5, < ADE >:< ED >= 32 − 5 = 27.
In this way, by discarding local unpromising item from each path local UP-Tree is constructed for each path.

5. Mathematical model

Let system S can be defined as:
$S = \{T, P, TU, TWU, RTU, DLU, UP - Tree\}$

T = set of transaction.

P = set of profit of per item.

TU = set of transaction utility per transaction.

TWU = set of transaction weighted utilities per transaction X.

RTU = set of recognized transaction utility.

DLU = set of itemset having highest promising value. UPtree = UPtree consist set of node,each node having utility count(support count).

$T = t1, t2, t_n$

$P = P1, P2, P_k$

Where, k is total number of items.

$TU = tu1, tu2, tu_n$

$TWU = tw1, tw2, tw_k$ $RTU = rt1, rt2, rt_p$ Where $p << k$

$DLU = x1, x2, x_z$

Function f_1 → takes the total transaction as input and calculate transaction utility.

$f_1 \rightarrow (t1, t2, TN) \rightarrow (tu1, tu2, tu_n)$
Function f_2 → reads transaction utility as input and calculate transaction weighted utilities.
$TU(T_d)$: Transaction utility of transaction T_d.
Function f_3 → Removes the unpromising itemset and generate re-organized transaction utility.
$f_3\ (TU, TWU) \mapsto (TU, TWU) \leq t, (r1, r2, r_n) \in R$.
Where t is user defined threshold.
Function f_4 → takes RTU as input and generate UPtree for frequent itemset.
$f_4\ (RTU) \rightarrow (r1, r2, .., r_p) \rightarrow (UP\,tree)$
Function f_5→ reads the UPtree and generates PHUI i.e.,potentially high utility itemset.
$f_5\ (UP\,tree) \rightarrow (phuI_1, phuI_2, .., phuI_k) \in PHUI$
Where,
PHUI is "Potentially high utility itemset of size 'k'.

6. Experimental evaluation

In this section expert examine the performance of proposed algorithm. The experiment has been carried out on a 2.40GHz, Intel core 3 processor is used with 4GB memory. The Windows 7 Operating System is used for examine purpose. Implementation of algorithm is done using Java language. Real datasets such as

FoodMart, chess utility, retail stores obtained from UCI repository of machine learning database.

Table 4: Time Comparison between Existing and Proposed Approach

Approach	RTU Time	RTU Path Time	PHUI Generation Time
Existing Approach	10 ms	16 ms	14 ms
Proposed Approach	5 ms	5 ms	4 ms

Table 4 shows that in proposed approach time required for calculating RTU (Reorganised Transaction Utility), Reorganised Path calculation and potential high utility itemset generation (PHUI) is less than existing algorithms.

Table 5: Time analysis between existing and proposed approach on various datasets

Dataset	Proposed Approach	Existing Approach
Data1000	1000 ms	2000 ms
Data5000	2000 ms	3000 ms
Data10000	3000 ms	5000 ms
Chess_utility_spmf	2000 ms	24000 ms
Foodmart	4000 ms	5000 ms
Retail_utility_spmf	900 ms	47000 ms

Table 5, shows the time (in ms) comparision between proposed approach and existing approach on various datasets. Experimental result shows that proposed system require less time to generate high utility itemsets than the existing system.

7. Performance comparison on different datasets

This part represents the performance comparison of proposed system with different datasets and different algorithms such as CHUI , D2HUp with MUP-Growth(Multithreaded Utility Pattern Growth) algorithm. As shown in figure 3 and figure 4 expert compare the performance of proposed approach with foodmart and retail utility store datasets with respect to time factor, it shows that proposed algorithm MUP-Growth performs better than other existing algorithm. The main reason behind this is that expert used multithreaded approach in existing UP-Growth algorithm to increase the efficiency of that existing algorithm. Fig. 5 shows the performance comparison under various datasets with proposed approach and previous approach. Proposed approach requires execution time less than previous approach time for the datasets. The data set which is test is in the text format stored in .txt format. For calculating experimental result following datasets are used,

1) Food Mart.
2) Chess Utility.
3) Retail Utility.

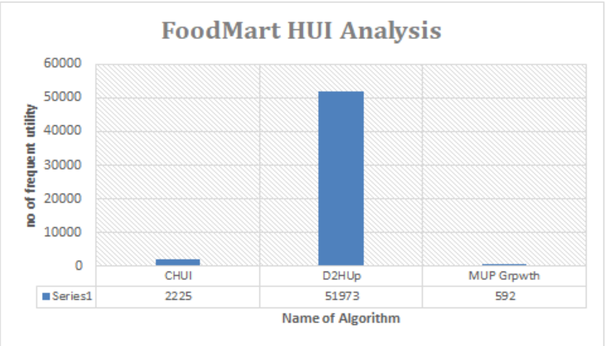

Fig. 3: Performance Analysis of Previous Algorithms and Proposed Algorithm on Food Mart Dataset.

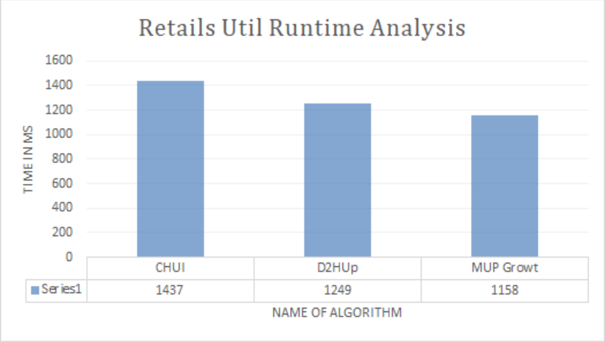

Fig. 4: Analyze HUI Algorithm on Retail Stores Dataset Along with Previous Algorithm.

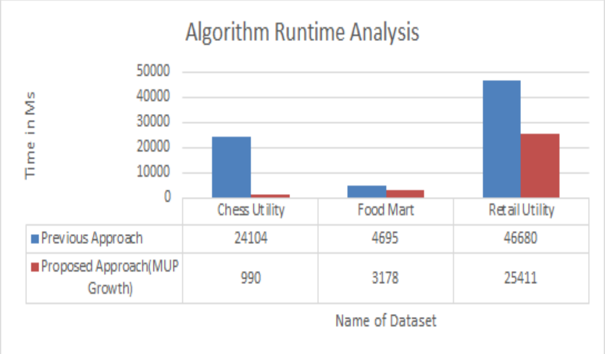

Fig. 5: Performance Comparison under Various Datasets with Proposed Algorithm.

8. Conclusion

In this paper, first we present the overview of big data techniques. An efficient algorithm MUP-Growth have proposed, which gives high utility item set in two scans and requires less time to produce result. It uses FP-Growth as a basic structure and uses tree data structure known as UP-Tree. It resolve the issue of FP-Growth, as there is only frequent item sets are generated but in case of MUP-Growth, it gives us frequent items also and from that it gives high utility items. It is useful to extract the profitable items from the transactional database and useful for storing only essential information in database. Moreover, we proposed some techniques which enhance the high utility mining performance. Experimental evaluation is carried out on various datasets to compute the performance evaluation. From performance measures it is concluded that proposed system improves the performance by reducing number of candidate itemset generation and also reduce the PHUI generation time.Such a techniques can be used in various fields like web usage mining, in large datasets to mine high utility items,etc.

Acknowledgement

We would sincerely like to thank our H.O.D Dr. Amol D. Potgantwar, Department of Computer Engineering, SITRC, and Nashik for his guidance, encouragement and the involvement shown in this research. His expert suggestions and feedback had immensely enhanced the effectiveness of this work.

References

[1] akesh Agrawal Ramakrishnan Shrikant*,"Fast Algorithms for Mining Association Rules", Proc. 20th Intl Conf. Very Large Data Base (VLDB), pp. 487-499, 1994.

[2] DanieleApiletti, ElenaBaralis, TaniaCerquitelli, PaoloGarza, FabioPul- virenti, LucaVenturini, "Frequent Itemsets Mining for Big Data: A Com- parative Analysis", 2017 Elsevier.

[3] Puneet Singh Duggal,Sanchita Paul,"Big Data Analysis: Challenges and Solutions",International Conference on Cloud, Big Data and Trust 2013, Nov 13-15, RGPV.

[4] Firat Tekiner1 and John A. Keane,"Big Data Framework", IEEE International Conference on Systems, Man, and Cybernetics 2013.

[5] Mrs. Mereena Thomas,"A Review paper on BIG Data", International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 02 Issue: 09, Dec-2015.

[6] Jaseena K.U. and Julie M. David,"Issuse, Challenges, And Solutions: Big Data Mining".

[7] Dhavapriya, N. Yasodha, "Big Data Analytics: Challenges and Solutions Using Hadoop, Map Reduce and Big Table ", International Journal of Computer Science Trends and Technology (IJCST) Volume 4 Issue 1, Jan - Feb 2016.

[8] Sabia and Love Arora,"Technologies to Handle Big Data: A Survey ".

[9] Ibrar Yaqooba, Ibrahim Abaker Targio Hashema, Abdullah Gania, Salimah Mokhtara,Ejaz Ahmeda, Nor Badrul Anuara, Athanasios V. Vasi- lakos ,"Big data:From beginning to future", International Journal of Information Man- agement 36 (2016) 12311247.

[10] Sravanthi,Tatireddy Subba Reddy, "Applications of Big data in Various Fields ", (IJCSIT) International Journal of Computer Science and Informa- tion Technologies, Vol. 6 (5) , 2015, 4629-4632.

[11] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Moal, and M.C. Hsu ," Mining Sequential Patterns by Pattern-Growth: The Prefixspan Approach ", IEEE Trans. Knowledge and Data Eng., vol.16, no.10, pp. 1424-1440, Oct. 2004.

[12] Aileen P. Wright a,*, Adam T. Wright b, Allison B. McCoy c, Dean F. Sittig d ,"The use of sequential pattern mining to predict next prescribed Medications ", Journal of Biomedical Informatics 53 (2015) 7380, 2014 Elsevier.

[13] Uma Dave, Jayna Shah, "Efficient Mining of High Utility Sequential Pattern from Incremental Sequential Dataset ", International Journal of Computer Applications (0975 8887) Volume 122 No.12, July 2015.

[14] Mrs. M.Kavitha 1, Ms.S.T.Tamil Selvi 2," Comparative Study on Apriori Algorithm and FP-Growth Algorithm with Pros and Cons ", International Journal of Computer Science Trends and Technology (I JCS T) Volume 4 Issue 4, Jul - Aug 2016.

[15] R. Chan, Q. Yang, and Y. Shen," Mining High Utility Itemsets ",Proc. IEEE Third Intl Conf. Data Mining, pp. 19-26, Nov. 2003. https://doi.org/10.1109/ICDM.2003.1250893.

[16] J. Han, J. Pei, and Y. Yin," Mining Frequent Patterns without Candidate Generation ", Proc. ACM-SIGMOD Intl Conf. Management of Data, pp. 1-12, 2000.

[17] Zhou, Zhiyong Zhong, Jin Chang, Junjie Li, Joshua Zhexue Huang, Shengzhong Feng, "Balanced Parallel FP-Growth with MapReduce ", 2012 IEEE.

[18] M.S. Mythili,A.R. Mohamed Shanavas, "Performance Evaluation of Apri- ori and FP-Growth Algorithms ", International Journal of Computer Appli- cations (0975 8887) Volume 79 No10, October 2013.

[19] Sankalp Mitra 1, Suchit Bande 2, Shreyas Kudale 3, Advait Kulkarni 4, Asst. Prof. Leena A. Deshpande 5,"Efficient FP-Growth Using Hadoop- (Improved Parallel FP-Growth) ", International Journal of Scientific and Research Publications, Volume 4, Issue 7, July 2014.

[20] Haoyuan Li, Yi Wang, Dong Zhang, Ming Zhang, Asst Edward Chang PFP: Parallel FP-Growth for Query Recommendation ", ACM.

[21] Hongjian Qiu, Rong Gu, Chunfeng Yuan, Yihua Huang*,"YAFIM: A Parallel Frequent Itemset Mining Algorithm with Spark ", 28th International Parallel & Distributed Processing Symposium Workshops, 2014 IEEE.

[22] Adetokunbo Makanju, Zahra Farzanyar, Aijun An, Nick Cercone,"Deep Parallelization of Parallel FPGrowth Using Parent-Child MapReduce", IEEE International Conference on Big Data, 2016.

[23] K.Dharmaraajan,M.A. Dorairangaswamy,"Analysis of FP-Growth and Apriori Algorithms on Pattern Discovery from Weblog Data",2016 IEEE International Conference on Advances in Computer Applications (ICACA). https://doi.org/10.1109/ICACA.2016.7887945.

[24] W. Wang, J. Yang, and P. Yu, "Efficient Mining of Weighted Association Rules (WAR) ", Proc. ACM SIGKDD Conf. Knowledge Discovery and Data Mining (KDD 00), pp. 270-274, 2000. https://doi.org/10.1145/347090.347149.

[25] K. Sun and F. Bai, "Mining Weighted Association Rules without Preassigned Weights ", IEEE Trans. Knowledge and Data Eng., vol. 20, no. 4, pp. 489-495, Apr. 2008. https://doi.org/10.1109/TKDE.2007.190723.

[26] F. Tao, F. Murtagh, and M. Farid, "Weighted Association Rule Mining Using Weighted Support and Significance Framework ", Proc. ACM SIGKDD Conf. Knowledge Discovery and Data Mining (KDD 03), pp. 661-666, 2003. https://doi.org/10.1145/956750.956836.

[27] A. Erwin, R.P. Gopalan, and N.R. Achuthan," Efficient Mining of High Utility Itemsets from Large Data Sets ", Proc. 12th Pacific-Asia Conf. Advances in Knowledge Discovery and Data Mining (PAKDD), pp. 554- 561, 2008. https://doi.org/10.1007/978-3-540-68125-0_50.

[28] H.F. Li, H.Y. Huang, Y.C. Chen, Y.J. Liu, and S.Y. Lee," Fast and Memory Efficient Mining of High Utility Itemsets in Data Streams ", Proc. IEEE Eighth Intl Conf. on Data Mining, pp. 881- 886, 2008.

[29] C.F. Ahmed, S.K. Tanbeer, B.-S. Jeong and Y.-K. Lee, "Efficient Tree Structures for High Utility Pattern Mining in Incremental Databases ", IEEE Trans. Knowledge and Data Eng., vol. 21, no. 12, pp.1708-1721, Dec. 2009. https://doi.org/10.1109/TKDE.2009.46.

[30] V.S. Tseng, C.-W. Wu, B.-E. Shie, and P.S. Yu,"UP-Growth: An Efficient Algorithm for High Utility Itemsets Mining ", Proc. 16th ACM SIGKDD Conf. Knowledge Discovery and Data Mining (KDD 10), pp. 253-262, 2010.

[31] B.-E. Shie, H.-F. Hsiao, V., S. Tseng, and P.S. Yu," Mining High Utility Mobile Sequential Patterns in Mobile Commerce Environments", Proc. 16th Intl Conf. DAtabase Systems for Advanced Applications (DASFAA 11), vol. 6587/2011, pp. 224-238, 2011.

[32] Vincent S. Tseng, Bai-En Shie, Cheng-Wei Wu, and Philip S. Yu, Fellow,"Efficient Algorithms for Mining High Utility Itemsets from Transactional Databases",IEEE Transactions on Knowledge And Data Engineering, VOL. 25, NO.8 AUGUST 2013.