Novel Utility Procedure for Filtering High Associated Utility Items from Transactional Databases

Srihari Varma Mantena, CVPR Prasad

***Abstract:** In data mining, mining and analysis of data from different transactional data sources is an aggressive concept to explore optimal relations between different item sets. In recent years number of algorithms/methods was proposed to mine associated rule based item sets from transactional databases. Mining optimized high utility (like profit) association rule based item sets from transactional databases is still a challenging task in item set extraction in terms of execution time. We propose High Utility based Association Pattern Growth (HUAPG) approach to explore high association utility item sets from transactional data sets based on user item sets. User related item sets to mine associated items using utility data structure (UP-tree) with respect to identification of item sets in proposed approach. Proposed approach performance with compared to hybrid and existing methods worked on synthetic related data sets. Experimental results of proposed approach not only filter candidate item sets and also reduce the run time when database contain high amount of data transactions.*

***Index terms:** Transactional databases, utility pattern, Association, high utility, utility pattern tree, item sets and data mining.*

I. INTRODUCTION

Exploring data from different data source like basket market related business organizations through different parametric aspects like attributes relations, item set relations present in transactional data sources. Business organizations arranged and kept item sets based on user knowledge with association relations.

Association rule mining is the method to mine or discover and analyze different item sets data relates to basket market organizations performed on transaction data sources. For efficient relation between data sources association rules are maintain threshold based support and confidence.

Revised Manuscript Received on August 30, 2019.

* Correspondence Author

Srihari Varma Mantena*, Research Scholar, Dept of CSE, Acharya Nagarjuna University, Guntur-522510, AP, INDIA, Email: vasista4u@gmail.com

Dr CVPR Prasad, Research Supervisor, Dept of CSE, Acharya Nagarjuna University, Guntur-522510, AP, INDIA, Email: prasadcvpr@gmail.com

Mining algorithms that can find the different association rules based on several types of attributes with respect to different types of transactions. To extract valuable information is often from unwanted association rule mining, which is suspected to presented users because of low support efficiency in transactional data sources.Useful data to be explored from different data with respect to declare different mining tasks such as frequent patterns based on different weights to mine high utility items. Frequent utility items to be mine which is present relates to mine frequent patterns from transactional data sets with respect to different items and other relations data sets like time series data bases [1], streaming databases [2] and mobile environment related databases [3].

Nevertheless, all the above approaches have some limitations, frequent pattern mining approach not consider relative importance of item sets. Weighted bases association rule mining approach proposed in [4], in this approach some of the items are infrequent if they have high utility, Quantity of item sets are also not considered, so that it fails to identify user interested items which contains profits, described based on weighted profit based on qualities. In this point of view utility mining is an emerging concept to identify high association item sets from transactional data sets. Mining utility of item set is interestingness while increasing profitability of item sets to different users. In transactional databases, utility of mining item sets consist two aspects: a) distinct item sets importance external utility, importance of items in transactions which is called internal utility of item sets. Utility of item to be calculated based on external utility and internal utility. If utility is less than user specified threshold minimum utility then item set called as High utility item set, otherwise called as low-utility item set. So that mining high associated rules based utility item sets is a challenging task used in wide range of marking related applications.It is not easy task to identify/mine associated rule based high utility item sets, mining high utility item set with search space is difficult if super item set of low item set consists high utility item set. To address these issues in mining associated high utility item sets, in this paper, propose High Utility based Association Pattern Growth (HUAPG) approach for mining high association based utility item sets with candidate pruning item sets. Candidate pruning item sets are explored using tree-based data structure (Utility-Pattern tree) with scan of database to identify candidate item sets used in proposed approach.

Major contributions of proposed approach is as follows

a) Implement the utility pattern procedure to mine associated high utility item sets, and maintain the importance of data relates to different utility patterns database proposed.

b) Decrease the over-estimated utilities of associated rule based high utility item sets, reduce the candidate item sets.

c) Performed experiments on different synthetic related data sets with existing approaches.

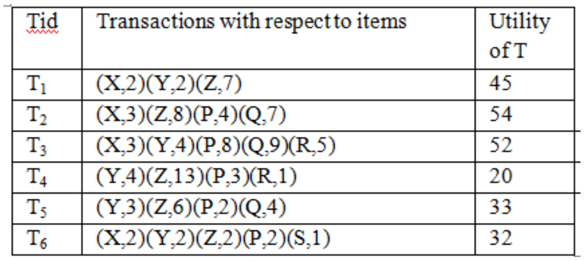
II. RELATED WORK

In this section, discuss different authors proposed approaches with respect to explore high utility item sets from transactional data sets. Also discuss high utility mining procedures with basic descriptions.

Interestingness quality measures speak to measurements during the time spent catching conditions and suggestions between database things, and express the quality of the example affiliation. Since regular item set age is considered as an costly task, mining successive shut item sets (starter thought

introduced in [4]) was proposed so as to describe the regular item sets. For instance, an item set X is indicated as shut successive item set if 6 9 item set

$\forall(itemset)\ X^{'} \supseteq X$ so that $t(X) = t(X')$. In this manner, the measurement of incessant shut item sets created is diminished in examination with the quantity of incessant item sets. The CLOSET calculation was proposed in [19] as another productive strategy for mining shut item sets. Storeroom utilizes a novel continuous example tree (FP-tree) structure, which is a compacted portrayal of the considerable number of exchanges in the database. Also, it utilizes a recursive gap and- vanquishes and database projection way to deal with mine long examples. These works are relates to traditional approaches worked on transactional data sets.



Tid	Transactions with respect to items	Utility of T
T_1	(X,2)(Y,2)(Z,7)	45
T_2	(X,3)(Z,8)(P,4)(Q,7)	54
T_3	(X,3)(Y,4)(P,8)(Q,9)(R,5)	52
T_4	(Y,4)(Z,13)(P,3)(R,1)	20
T_5	(Y,3)(Z,6)(P,2)(Q,4)	33
T_6	(X,2)(Y,2)(Z,2)(P,2)(S,1)	32

III. BASIC PRELIMINARIES USED IN UTILITY MINING

This section describes the basic parameters used in proposed approach with different notations. Let us consider different finite item sets $I = \{t_1,\ t_2\ ,t_3\ ,.....,t_m\}$ in that each item present in node consists unit profit $pr(i_p)$ which is present in between 1 to m. X be the set of distinct item sets $\{di_1, di_2, di_3,....., di_k\}$ where i values present in between 1 to k, k is the length of distinct item sets. Let D be the transactional data base $D = (T_1, T_2\ ,.......,T_n)$ which is the combination of different transactions, each transaction describe a single identifier d, which is represented with t_{id}. Each node item associated with quantity $q(i_p\ ,T_d\ )$ which is the item set quantity. We describe preliminaries relates to different transactions with following used definitions based on descriptions of available in table

1 (Description about transactions and their utility) with profits.

Def-1: $u\left(i_p\,,T_d\right)$be the item i utility with different transaction t_d and profit pr is described as $pr(i_p)\times q(i_p\,,T_d\,)$

Def-2: $u\ (X\ ,T_d)$be the item set utility X with t_d and extensive profitability as described as

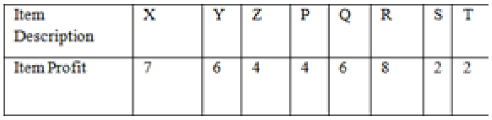
IV. HIGH UTILITY BASED ASSOCIATION PATTERN GROWTH PROCEDURE

$$\sum_{i_p\in X \wedge X\subseteq T_d} u\left(X,T_d\right)$$

This section describes the procedure of high utility association utility pattern with respect to

Def-3: Based on def 1, def 2, if any item should be called as high association item then it is satisfied minimum utility of transactions with profitability.

Table 1, 2: Example of transactional database, transactions with their profits**.**



Item Description	X	Y	Z	P	Q	R	S	T
Item Profit	7	6	4	4	6	8	2	2

$$u(\{X\ \},T_1\ ) = 5\times1 = 5;$$

$$u(\{XP\},T_1)=u(\{X\},T_1)+u(\{P\},T_1)=5+2=7;$$

transactional data sets. Proposed approach mainly consist three steps in processing of high association utility item sets. 1) In first step, scan the data base (which consists different transactional data items) using utility pattern tree 2) calculate potential association utility item sets from scanned data 3) Finally optimize the filtered and high association item sets.

The term used in our approach i.e. potential association high utility item sets (PAHUI) taken from high transaction utility used in traditional approaches.

A. UP-tree: Data structure for Scanning Transactional data base

Using this data structure mining the performance and eliminate scanning original transactional database and also maintain data with different transactions and high association utility items. In this scenario, we introduce two strategies to minimize utilities to be overestimated stored in global utility pattern tree structure. First we describe basic elements used in UP-tree, and then describe overall procedure to construct and running based on table 1

$$u(\{XP\})=u(\{XP\},T_1)+u\{XP\},T_3)+u(\{XP\},T_6$$
$$=7+22+7=36$$

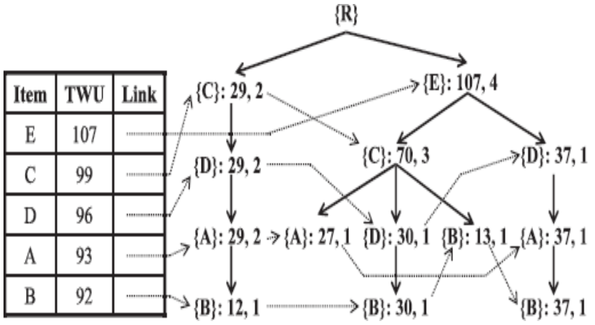
Based on the above example, min_util=30 then {XP}

example.

i. UP-Tree Elements

be the high utility item set. Problem behind this example is, given data base P and threshold min_util specified by user, main problem is mining high association rule based utility items from transaction data base d is to identify item sets relates to complete data which are utilities are higher than selected min- util. So that, transaction weighted high association measure is introduced.

Def 4: (transaction weighted high association measure) if G be the complete item set, and it is not high potential transactional weighted utility, any subset of G be the item set describes low utility from overall transactional data sets.



Item	TWU	Link
E	107	
C	99	
D	96	
A	93	
B	92	

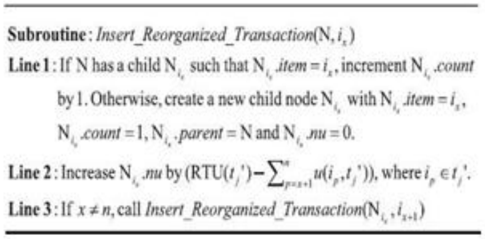
Fig.1. Integrated high utility item set with respect to min_util=33.3

To describe tree-traversal conditions of UP- tree to arrange attributes with table named as table_header, table present in header, check item records with name of the item, utility that can be overestimated with node link. Occurrence of node utility which is present in utility tree and nodes present in tree have same efficiency.

ii. Utilities of Decreasing Global Node while constructing UP tree

Mining potential high utilities present in item sets, divide and conquer procedure can be used in tree construction based on processing item sets. In this scenario, each search space can be divided into sub spaces procedures shown in figure 1. As shown in fig 1, it can be divide with following sub spaces.

automatically deleted descendent nodes to other node utilities present in UP-tree. Process to construct UP- tree to be performed. Based on DGN procedure, node utilities which are very close or not close to utility tree and it is to be reduced in improvement. DGN is the approach to reduce descendent nodes which are not satisfy min-util conditions and maintain long transactions. Based on DGN, constructing UP-tree as follows:



```
Subroutine: Insert_Reorganized_Transaction(N, $i_x$)
Line 1: If N has a child $N_{i_x}$ such that $N_{i_x}.item = i_x$, increment $N_{i_x}.count$
        by 1. Otherwise, create a new child node $N_{i_x}$ with $N_{i_x}.item = i_x$,
        $N_{i_x}.count = 1$, $N_{i_x}.parent = N$ and $N_{i_x}.nu = 0$.
Line 2: Increase $N_{i_x}.nu$ by $(RTU(t_j') - \sum_{p=x+1}^{n} u(i_p, t_j'))$, where $i_p \in t_j'$.
Line 3: If $x \neq n$, call Insert_Reorganized_Transaction($N_{i_x}$, $i_{x+1}$)
```

Fig.2. Procedure of *Insert_Reorganized_Transaction*

UP-tree data structure is constructed based on scan of transactional database, in first scan of transactional utility to be calculated at same time, and also computed transactional weighted utility for each item, after processing all the items apply DGN. In this scenario, item sets are re-organized by promising and un-promising items and then arrange all the items in sorting order. Using *Insert_Reorganized_Transaction* is used for constructing UP-tree based on some subroutines shown in figure 2. When transaction can be re-organized i.e. $t'_j = \{t_1, t_2, ..., t_n\}(i_k \in I, 1 \le k \le n)$ and store in utility tree then *Insert _ Reorganized _ Transaction* is called for N number of nodes, recursively this

{B}- Conditional tree with different notations

{A}- Tree w/t {B}

{D}- w/t {B} & {A}

{C}- w/t {B}{A} & {D}

{E}- w/t {B},{A},{D} & {C}

Closely it can be observed that, all the paths not related to {B} of {A}, since node behind {B} present in {A} in global space in UP-tree In other node representations, the items which nodes are related descendent item arrange into node i does not arrange in tree, only ancestor node will appear in $\{i_m\}$. Based on these criteria, proposed UP-tree DGN procedure

procedure continuous for last node present in tree. After transactions re-organized tree representation can be shown in

After inserting all the items with their transactions global tree can be constructed. Compared with figure 1 utilities of nodes values to be described effectively by DGN.

B. Associated Pattern Growth Procedure

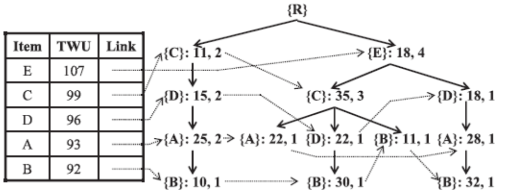
Pattern growth procedure present in UP-tree gives better and high performance than traditional approaches to decrease over-estimated utility item sets. To optimize the over-estimated results, we propose associated pattern growth procedure (APGP). In UP-tree transactional utilities discussed in above section used *min_util* to reduce minimum utility where as APGP use *minimal_node_utilities* in link between node and its associated items to make estimated pruning values based on available items in database. Basic procedure used for mining high associated item sets described in algorithm 1 with different notations.

Algorithm 1: Procedure for associated high utility pattern items.

```
Input: Data in terms D= {d1,d2,....dn}, min_util, user_defined_parameters.
Output: Associated high transactional utilities for different item sets.
    1. Processing data set based on initialized item sets stored in tree_node,T_node=0
    2. For each item set calculate trans_utility using based on parameters
    3. Calcu Potential_transactional_itemset weight
    4. Update tree every time in processing of data item sets with different relations.
    5. Maximize the potential_utility and reduce based on updated TP_tree
    6. Minimize the high associated itemsets with different notations.
Optimize high confidence item sets with relations.
```

As shown in algorithm 1, Node minimal utility can be acquired from transactional data sets by constructing high UP-tree structure. Basic element i.e root element with minimum utility i.e. N_{min} for each node present in utility tree, minimum utility of node ($N._{mnu}$) with item N.name

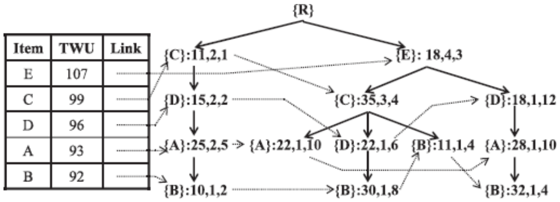
figure 3.



Item	TWU	Link
E	107	
C	99	
D	96	
A	93	
B	92	

Fig.3. UP-tree after applying DGN on item sets.

with different transactional item sets, if $N._{mnu}$ is greater than $u(N.name,T_{current})$ and N_{mnu} is the combination to different node $u(N.name,T_{current})$.



Item	TWU	Link
E	107	
C	99	
D	96	
A	93	
B	92	

Fig.4. Minimal node utilities tree representation with associated utility pattern tree.

As shown in figure 4, it describe the N.mnu is the global minimal utility at each node, it repeats until it reach last node describe in figure 4.

i. Mine Efficient Associated high utility Item sets

After identification of potential utility of different item sets, mine association high utility item sets with respect to potential weighted utility item sets by scanning of transactional database. In our proposed framework, utilities of potential utility itemsets is smaller than or equal to transactional weighted utility by reducing the performance of proposed approach whereas in second scan of database the our approach gives better and efficient high associated utility itemsets. Based on these criteria, we reduce execution cost, I/O cost with respect to memory, CPU processing usage for processing item sets in transactional data sets.

V. RESULTS

This section describes proposed approach performance with respect to processing transactional data sets. Experiments done in our approach performed on I3 processor, 4GB RAM and Window operating systems with technical software's like JAVA and NETBEANS. Algorithm implemented in this approach using JAVA libraries performed on synthetic data sets.

These synthetic data sets are generator by data generator done in implementation of proposed approach with respect process transactional high associated utility item sets shown in figure 5.

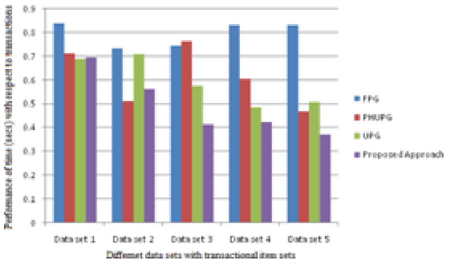
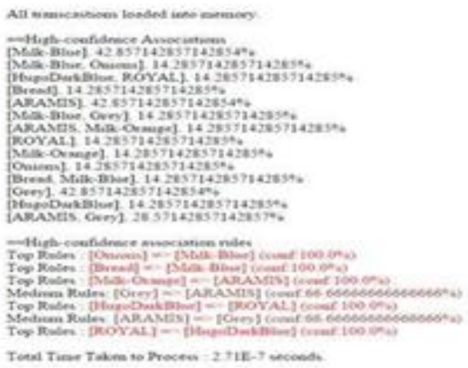


Fig.5. High associated transactional utilities from transactional data sets.

Figure 5 shows the performance of time of the proposed approach in comparison of previous approaches i.e. frequent pattern growth (FPG), potential high utility pattern growth (PHUPG), and utility pattern growth (UPG).

Execution time for processing item sets: To show the performance of proposed approach with respect to different parameter accessing from synthetic data sets. Execution for different approach with different item sets with data sets as follows:



```
All transactions loaded into memory

==High-confidence Associations
[Milk-Blue], 42.857142857142854%
[Milk-Blue, Onions], 14.285714285714285%
[HugoDarkBlue, ROYAL], 14.285714285714285%
[Bread], 14.285714285714285%
[ARAMIS], 42.857142857142854%
[Milk-Blue, Grey], 14.285714285714285%
[ARAMIS, Milk-Orange], 14.285714285714285%
[ROYAL], 14.285714285714285%
[Milk-Orange], 14.285714285714285%
[Onions], 14.285714285714285%
[Bread, Milk-Blue], 14.285714285714285%
[Grey], 42.857142857142854%
[HugoDarkBlue], 14.285714285714285%
[ARAMIS, Grey], 28.571428571428573%

==High-confidence association rules
Top Rules : [Onions] ==> [Milk-Blue] (conf:100.0%)
Top Rules : [Bread] ==> [Milk-Blue] (conf:100.0%)
Top Rules : [Milk-Orange] ==> [ARAMIS] (conf:100.0%)
Medium Rules: [Grey] ==> [ARAMIS] (conf:66.66666666666666%)
Top Rules : [HugoDarkBlue] ==> [ROYAL] (conf:100.0%)
Medium Rules: [ARAMIS] ==> [Grey] (conf:66.66666666666666%)
Top Rules : [ROYAL] ==> [HugoDarkBlue] (conf:100.0%)

Total Time Taken to Process : 2.71E-7 seconds.
```

Fig.6. Performance of time based on different transactions

In figure 6, it describe the performance of execution of different approaches, by observing this proposed approach gives better time results compared to existing approaches applied on synthetic data sets.

Memory utilization for processing transactions: Kernel functions present in windows to be performed and processing different data relations based on memory representation to calculate the relations.

to processing of transactional data sets. User interface design to

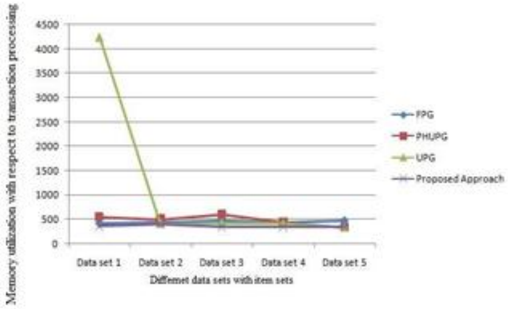


Fig 7 Performance of memory for different item sets processing.

Figure 7 describes the proposed approach performance in terms of memory usage to process different transactional data sets with sequential processing items.

CPU input and output cost for processing items: Input and out processing cost in terms of usage of CPU to evaluate the transactions from transactional data sets with different notations.

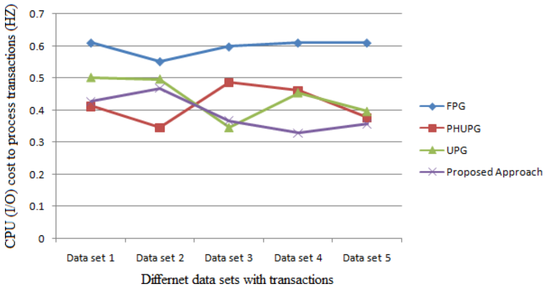


Fig.8. Performance of I/O cost for item sets

Figure 8 describe CPU usage with I/O (input/output) to process and describe relevant features with respect to transaction to find high associated transactional utilities from synthetic data related transactions. Finally with respect to different parameters, proposed approach gives better and efficient results with comparison to existing approaches.

VI. CONCLUSION

In this paper, we present Utility based Association Pattern Growth (UAPG) methodology to explore association based high utility item sets based on pruning of user item sets. To store and manage high association utility item sets using tree based data structure in transactional databases. By using this data structure, successfully calculate high potential association utility items from transactional data sets and increase the performance association utility mining from transactional data sets.

We performed out approach on synthetic or real time datasets with extensive attribute representation. Finally experiments of proposed approach gives better and increase the efficiency in term of time, memory and other parameters with comparison to existing algorithms. Further improvement of research is to outsource user's transactional data into different outsourced and data sharing organizations then privacy is basic measure handle transactionssecurely.

REFERENCES

1. M.Y. Eltabakh, M. Ouzzani, M.A. Khalil, W.G. Aref, and A.K. Elmagarmid, "Incremental Mining for Frequent Patterns in Evolving Time Series Databases," Technical Report CSD TR#08-02, Purdue Univ., 2008.
2. S.C. Lee, J. Paik, J. Ok, I. Song, and U.M. Kim, "Efficient Mining of User Behaviors by Temporal Mobile Access Patterns," Int'l J. Computer Science Security, vol. 7, no. 2, pp. 285-291, 2007.
3. C.H. Lin, D.Y. Chiu, Y.H. Wu, and A.L.P. Chen, "Mining Frequent Itemsets from Data Streams with a Time-Sensitive Sliding Window," Proc. SIAM Int'l Conf. Data Mining (SDM '05), 2005.
4. 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.
5. C.H. Cai, A.W.C. Fu, C.H. Cheng, and W.W. Kwong, "Mining Association Rules with Weighted Items," Proc. Int'l Database Eng. and Applications Symp. (IDEAS '98), pp. 68-77, 1998.
6. Dam, T.L., Li, K., Fournier-Viger, P., Duong, Q.H.: An efficient algorithm for mining top-rank-k frequent patterns.Applied Intelligence 45(1), 96–111 (2016)
7. Dam, T.L., Li, K., Fournier-Viger, P., Duong, Q.H.: CLS-Miner: efficient and effective closed high utility itemset mining. Frontiers of Computer Science pp. 1–27 (2017)
8. Dam, T.L., Li, K., Fournier-Viger, P., Duong, Q.H.: An efficient algorithm for mining top-k on-shelf high utility itemsets. Knowledge and Information Systems pp. 1–35 (2017)
9. Duong, Q.H., Liao, B., Fournier-Viger, P., Dam, T.L.: An efficient algorithm for mining the top-k high utility itemsets, using novel threshold raising and pruning strategies. Knowledge-Based Systems 104, 106–122 (2016)
10. Fournier-Viger, P., Gomariz, A., Gueniche, T., Soltani, A., Wu, C.W., Tseng, V.S.: SPMF: A java open-source pattern mining library. Journal of Machine Learning Research 15, 3569–3573 (2014)
11. Fournier-Viger, P., Lin, J.C.W., Duong, Q.H., Dam, T.L.: PHM: Mining Periodic High-Utility Itemsets. In: Lecture Notes in Computer Science, ICDM 2016, pp. 64–79. Springer (2016)
12. Fournier-Viger, P., Wu, C.W., Zida, S., Tseng, V.: FHM: Faster High-Utility Itemset Mining Using Estimated Utility Co-occurrence Pruning. In: Foundations of Intelligent Systems, Lecture Notes in Computer Science, vol. 8502, pp. 83–92. Springer International Publishing (2014).
13. J.H. Chang, "Mining Weighted Sequential Patterns in a Sequence Database with a Time- Interval Weight," Knowledge-Based Systems, vol. 24,no. 1, pp. 1-9, 2011.
14. C. K.-S. Leung and F. Jiang, "Frequent Item set Mining of Uncertain Data Streams using the Damped Window Model," in Proc. of the 26th Annual ACM Symposium on Applied Computing, pp. 950-955, Taichung, Taiwan, March, 2011
15. Lee, S., Park, J.S.: Top-k high utility itemset mining based on utility-list structures. In: 2016 International Conference on Big Data and Smart Computing (BigComp), pp. 101–108 (2016)
16. Lin, J.C.W., Gan, W., Fournier-Viger, P., Hong, T.P., Tseng, V.S.: Efficient algorithms for mining high-utility itemsets in uncertain databases. Knowledge-Based Systems 96, 171 – 187 (2016).
    Sahoo, J., Das, A.K., Goswami, A.: An efficient fast algorithm for discovering closed+ high utility itemsets. Applied Intelligence pp. 1–31 (2016).

AUTHORS PROFILE

Srihari Varma Mantena is a research scholar in ANU University, Guntur. His area of research is Data Mining.

Dr CVPR Prasad is presently working as HOD, Dept. of CSE, Malla Reddy Engineering College for Women, Hyderabad. His areas of research intrest are Data Mining and Machine Learning.