International Journal of Technology Research and Management
ISSN (Online): 2348-9006
Vol 5 Issue 1 January 2018

An Enhanced Technique for Mining Top K High Utility Items from a Data Set

Yogesh Vashishtha[1], Abhishek Raghuvanshi[2]
Department Of IT, MIT , Ujjain(MP), 456010, India[1, 2]
Yogeshvashishtha09@gmail.com[1], Abhishek14482@gmail.com[2]

Abstract: *High utility thing sets (HUIs) mining is a rising subject in information mining.HUIs determined least utility edge min utility. Notwithstanding, setting min utility suitably is a troublesome issue for clients. As a rule, finding a fitting least utility edge by experimentation is a monotonous procedure for clients. In the event that min utility is Set too low, an excessive number of HUIs will be produced, which may bring about the mining procedure to be exceptionally wasteful. Then again, if min utility is set too high, it is likely that no HUIs will be found. In this paper, we address the above issues by proposing another structure for top-k high utility thing set mining, where k is the coveted number of HUIs to be mined. Two sorts of proficient calculations named TKU (mining Top-K Utility thing sets) and TKO (mining Top-K utility thing sets in one stage) are proposed for mining such thing sets without the need to set min_utility. We give an auxiliary examination of the two calculations with talks on their preferences and restrictions. Exact assessments on both genuine and manufactured datasets demonstrate that the execution of the proposed calculations is near that of the ideal instance of best in class utility mining calculations.*

Keywords: *Utility mining, high utility item set mining, top-k pattern mining, top-k high utility item set mining, Data mining ,frequent itemset, transactional database.*

1. INTRODUCTION

FREQUENT item set mining (FIM) is a fundamental research topic in data mining. However, the traditional FIM may discover a large amount of frequent but low-value item sets and lose the information on valuable item sets having low selling frequencies. Hence, it cannot satisfy the requirement of users who desire to discover item sets with high utilities such as high profits. To address these issues, utility mining emerges as an important topic in data mining and has received extensive attention in recent years. In utility mining, each item is associated with a utility (e.g. unit profit) and an occurrence count in each transaction (e.g. quantity). The utility of an item set represents its importance, which can be measured in terms of weight, value, quantity or other information depending on the user specification. An item set is called high utility item set (HUI) if its utility is no less than a user-specified minimum utility threshold min_ utility. HUI mining is essential to many applications such as streaming analysis, market analysis, mobile computing and biomedicine. However, efficiently mining HUIs in databases is not an easy task because the downward closure property used in FIM does not hold for the utility of item sets. In other words, pruning search space for HUI mining is difficult because a superset of a low utility item set can be high utility. To tackle this problem, the concept of Transaction Weighted Utilization (TWU) model [13] was introduced to facilitate the performance of the mining task. In this model, an item set is called high transaction-weighted utilization item set (HTWUI) if its TWU is no less than min_ utility, where the TWU of an item set represents an upper bound on its utility. Therefore, a HUI must be a HTWUI and all the HUIs must be included in the complete set of HTWUIs. A classical TWU model-based algorithm consists of two phases. In the first phase, called phase I, the complete set of HTWUIs are found. In the second phase, called phase II, all HUIs are obtained by calculating the exact utilities of HTWUIs with one database scan. Although many studies have been devoted to HUI mining, it is difficult for users to choose an appropriate minimum utility threshold in practice. Depending on the threshold, the output size can be very small or very large. Besides, the choice of the threshold greatly influences the performance of the algorithms. If the threshold is set too low, too many HUIs will be presented to the users and it is difficult for the users to comprehend the results. A large number of HUIs also causes the mining algorithms to become inefficient or even run out of memory, because the more HUIs the algorithms generate, the more resources they consume. On the contrary, if the threshold is set too high, no HUI will be found. To find an appropriate value for the min_utility threshold, users need to try different thresholds by guessing and re-executing the algorithms over and over until being satisfied with the results. This process is both inconvenient and time-consuming.

2. OBJECTIVE AND SCOPE

UP-Growth performs better only when min_utility is UP-Growth outperforms UP-Growth although they have tradeoffs on memory usage. The reason is that UP-Growth+ utilizes minimal node utilities for further decreasing overestimated utilities of item sets. Even though it spends time and memory to check and store minimal node utilities, they are more effective especially when there are many longer transactions in databases. In contrast, UP-Growth performs better only when min_ utility is small. This is because when number of candidates of the two algorithms is similar, UP-Growth+ carries more computations and is thus slower. Finally, high utility item sets are efficiently identified from the set of PHUIs which is much smaller than HTWUIs generated by IHUP. By the reasons mentioned above, the proposed algorithms UP-Growth and UP-Growth+ achieve better performance than IHUP algorithm.til is small. This is because when number of candidates of the two algorithms is similar, UP-Growth+ carries more computations and is thus slower. Finally, high utility item sets are efficiently identified from the set of PHUIs which is much smaller than HTWUIs generated by IHUP. By the reasons mentioned above, the proposed algorithms UP-Growth and UP-Growth +achieve better performance than IHUP algorithm.

3. LITERATURE SURVEY

[1] Vincent S. Tseng, Bai-En Shie, Cheng-Wei Wu, and Philip S. Yu, "Efficient Algorithms for Mining High Utility Item sets from Transactional Databases", IEEE computer society, Aug 2013.
Seed Idea:- This paper proposes a high utility item set growth approach that works in tree structured algorithm. Incur problem of producing large no. of candidate item sets for HUI. Large number of candidate item sets degrades the mining performance in terms of execution time and space requirement. The situation may become worse when the database contains lots of long transactions or long high utility item sets. In this paper, we propose two algorithms, namely utility pattern growth (UP-Growth) and UP-Growth+, for mining high utility item sets with a set of effective strategies for pruning candidate item sets. The information of high utility item sets is maintained in a tree-based data structure named utility pattern tree (UP-Tree) such that candidate item sets can be generated efficiently with only two scans of database.

[2]Cheng-Wei Wu, Philippe Fournier-Viger, Philip S. Yu, Fellow, IEEE, Vincent S. Tseng, "Efficient algorithms for mining the concise and lossless representation of high utility item sets," IEEE Trans. Knowl. Data Eng., vol. 27, no. 3, pp. 726–739, Mar. 2014.
Seed Idea:- We propose three efficient algorithms named AprioriCH (Apriori-based algorithm for mining High utility Closed+ itemsets), AprioriHC-D (AprioriHC algorithm with Discarding unpromising and isolated items) and CHUD (Closed+ High Utility itemset Discovery) to find this representation. Further, a method called DAHU (Derive All High Utility itemsets) is proposed to recover all HUIs from the set of CHUIs without accessing the original database. Results on real and synthetic datasets show that the proposed algorithms are very efficient and that our approaches achieve a massive reduction in the number of HUIs. In addition, when all HUIs can be recovered by DAHU, the combination of CHUD and DAHU outperforms the state-of-the-art algorithms for mining HUIs.

[3] Miss. A. A. Bhosale , S. V. Patil, Miss. P. M. Tare, Miss. P. S. Kadam"High Utility Item setsMining on Incremental Transactions using UP-Growth and UP-Growth+ Algorithm":
seed Idea: Effective disclosure of item sets with high utility like benefits manages the mining high utility item sets from an exchange database Although various important methodologies have been proposed as of late, these calculation acquire the issue of creating an extensive number of competitor item sets for high utility item sets and most likely debases the mining execution as far as execution time and memory space. In this paper, we propose two calculations, viz., utility example development (UP-Growth) and Improved UP-Growth i.e. Enhanced Utility Pattern Growth, for mining high utility item sets with an arrangement of successful systems for pruning applicant item sets. The data of high utility item sets is kept up in a smaller

tree-based information structure utility example tree (UP-Tree), it filter the first database twice to oversee information organized way. Proposed calculations, particularly Improved UP Growth, decrease the quantity of applicants adequately as well as outflank different calculations considerably as far as runtime and memory utilization, particularly when databases contain loads of long exchanges.
Limitaton: Mining is only used in “Transactions” Data Set.

[4] Komal Surawase, Madhav Ingle “Mining Highly Utilized Item Set from Transaction Database”:
Seed Idea: Mining exceptionally used thing sets from a value-based dB intends to find the thing sets with high utility as benefits. In spite of the fact that various Algorithms have been created yet they bring about the issue as it produce huge arrangement of applicant Item sets likewise require number of database output. In regular thing set mining the unit benefits and bought amounts of the things are not taken into contemplations and weighted mining benefit is not viewed as just weight is to be considered. Expansive number of Item sets decreases the execution of mining as for execution time and space prerequisite. At the point when database contains countless this circumstance becomes more awful. In proposed framework for make UP-tree and UP-tree mining calculations named as Up-Growth and Improved Up-Growth the data of very used thing sets is recorded in tree based information structure called Utility Pattern Tree which is a minimal tree representation of things in exchange database. With the assistance of Utility Pattern Tree, applicant thing sets produced inside just two sweeps of the database. Proposed calculations not just decrease various competitor thing sets additionally spare memory and time.
Limitation: it generates huge set of Potential High Utility Item sets.

[5]Liang Wang, David Wai-Lok Cheung, Reynold Cheng, Member, IEEE, Sau Dan Lee, and Xuan S. Yang, “Efficient Mining of Frequent Item Sets on Large Uncertain Databases”, IEEE Transact ions On Knowledge And Data Engineering, volume 24, Issue No. 12, pp 2170-2183, DECEMBER 2012.
Seed Idea: The data handled in emerging applications like location-based services, sensor monitoring systems, and data integration, are often inexact in nature. In this paper, we study the important problem of extracting frequent item sets from a large uncertain database, interpreted under the Possible World Semantics. This issue is technically challenging, since an uncertain database contains an exponential number of possible worlds. By observing that the mining process can be modeled as a Poisson binomial distribution, we develop an approximate algorithm, which can efficiently and accurately discover frequent item sets in a large uncertain database. We also study the important issue of maintaining the mining result for a database that is evolving (e.g., by inserting a tuple). Specifically, we propose incremental mining algorithms, which enable probabilistic frequent item set results to be refreshed. This reduces the need of re-executing the whole mining algorithm on the new database, which is often more expensive and unnecessary. We examine how an existing algorithm that extracts exact item sets, as well as our approximate algorithm, can support incremental mining.
Limitation: uncertain dataset.

4. EXISTING SYSTEM MECHANISM

FREQUENT item set mining is a fundamental research topic in data mining (FIM) mining. However, the traditional FIM may discover a large amount of frequent but low-value item sets and lose the information on valuable item sets having low selling frequencies. Hence, it cannot satisfy the requirement of users who desire to discover item sets with high utilities such as high profits. To address these issues, utility mining emerges as an important topic in data mining and has received extensive attention in recent years. In utility mining, each item is associated with a utility (e.g. unit profit) and an occurrence count in each transaction (e.g. quantity). The utility of an item set represents its importance, which can be measured in terms of weight, value, quantity or other information depending on the user specification. An item set is called high utility item set (HUI) if its utility is no less than a user-specified minimum utility threshold min_ utility. HUI mining is essential to many applications such as streaming analysis, market analysis, mobile computing and biomedicine. Efficiently mining HUIs in databases is not an easy task because the downward closure property used in FIM does not hold for the utility of item sets. In other words, pruning search space for HUI mining is difficult because a superset of a low utility item set can be high utility.

5. PROPOSED SYSTEM MECHANISM

The concept of transaction weighted utilization (TWU) model was introduced to facilitate the performance of the mining task. In this model, an item set is called high transaction-weighted utilization item set (HTWUI) if its TWU is no less than min_utiity l, where the TWU of an item set represents an upper bound on its utility. Therefore, a HUI must be a HTWUI and all the HUIs must be included in the complete set of HTWUIs. A classical TWU model-based algorithm consists of two phases. In the first phase, called phase I, the complete set of HTWUIs are found. In the second phase, called phase II, all HUIs are obtained by calculating the exact utilities of HTWUIs with one database scan. Two efficient algorithms named TKU (mining Top-K Utility items etc.) and TKO (mining Top-K utility item sets in one phase) are proposed for mining the complete set of top-k HUIs in databases without the need to specify the min_ utility threshold. The construction of the UP-Tree and prune more unpromising items in transactions, the number of nodes maintained in memory could be reduced and the mining algorithm could achieve better performance.

Proposed Algorithms

- TKO (Top k in one phase)
- TKU (Top k in Utility)

Input: All HUI tree T_s and header tables Hs in the current window, an item set based item set (base –item set is initialized as null), as list TKValueList , minimum utility value min_uti.

Output: TKHUIs

Begin

(1) Find top-k maximal total utility value of itemset in Hs to TKValueList
(2) Add a field add-information to each leaf-node
(3) For each item Q in HL do from the last item of HL and HL is one HS
//Step 1:Calculate utility information of the node Q
(4) Float twu=0,BU=0,SU=0,NU=0;
(5) For each header table H in Hs do
(6) For each node N for the item Q in the tree T corresponding to H do
(7) BU+=T.N.bu;
(8) SU+=T.N.su;
(9) NU+=T.N.nu;// N.nu is a utility for item Q in the list N.piu
(10) End For
(11) End For
(12) twu=BU+SU
//Step 2 :Generate new itemset and create new sub tree and header table
(13)If(twu>= minUti) then
(14) base-itemset={Q};
(15) create a sub HUI tree subT and a header table subH for base-itemset ;
(16) sub-Mining(SubT,SubTbase-itemset,TKValueList,min_uti);
(17)Remove item Q from itemset base-itemset;
(18)End If
// Step 3:Pass add-information on node Q to parent node
(19) Move each node's bac-info to its parent;
(20) End For
(21) Delete item set whose value are less than minUti from TKHUIs;
(22) Return TKHULS:
END

Advantages of algorithms:

1. **TKO (Top K in one Phases)**
   - min_util threshold is not given in advance in top-k HUI mining.
   - Process of Algorithms is fast as compared to other algorithms.
2. **TKU (Top k in Utility)**
   - Result of High utility pattern are Correct
3. **Up Growth**
   - Calculates more sets of frequent items
   - Build a compact data Structure called Up- tree.
4. **Pattern Mining**
   - Data Base id Scanned only one time.
   - It required less memory.
   - Highly suitable for interactive mining.

Advantages of Proposed System:

- Two efficient algorithms named TKU (mining Top-K Utility itemsets) and TKO (mining Top-K utility item sets in one phase) are proposed for mining the complete set of top-k HUIs in databases without the need to specify the min_util threshold
- The proposed algorithm has less search space so it needs less memory
- It scans the database only once
- It is easy to implement
- Its performance is good in dense database

6. ARCHITECTURE

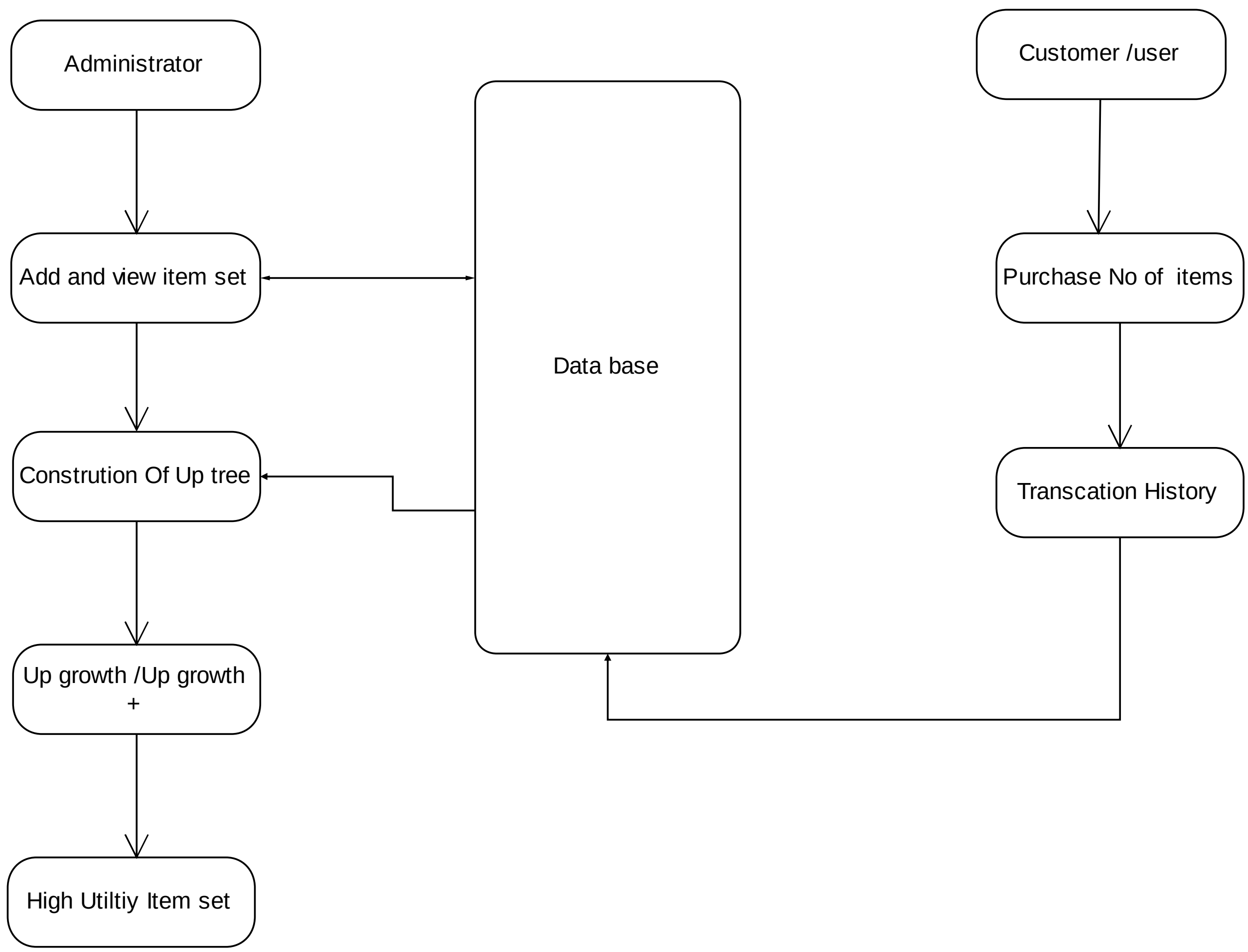


Fig 1: System Architecture

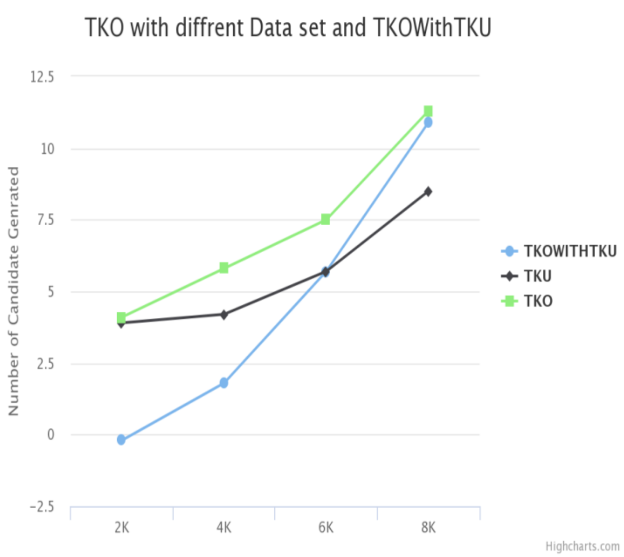
7. RESULT

As Result Top K Algorithm is applied on the Different Data set such as the dataset such as the Mushroom, Chess and Accidents with varied k respectively. In this Graph TKUWITHTKO is best performance among top-k HUI mining algorithms, in this graph TKO with TKU Spend 120 almost time to seconds to complete the mining process while REPT and TKU take more than 160 seconds and TKO 70 Seconds. In comparison Table TKO and TKU Algorithms is compared with different parameter such as the TWU and CHUD and Find garbage values.

8. COMPARISON BETWEEN ALGORITHMS

Sr.No	PARAMETRE	TKO	TKU
01	Anti-monotonicity	Yes	No
02	Fallow TWU high utilitypattern	No	Yes
03	CHUD(Closet high utility	Yes	Yes

	dataset)		
04	Finding Garbage Value in algorithm	Yes	No
05	Min until value	==0	==0



9. CONCLUSION

In this paper, we have studied the problem of top-k high utility item sets mining, where k is the desired number of high utility item sets to be mined. Two efficient algorithms TKU (mining Top-K Utility item sets) and TKO (mining Top-K utility item sets in One phase) are proposed for mining such item sets without setting minimum utility thresholds. TKU is the first two-phase algorithm for mining Top-k high utility item sets, which incorporates five strategies PE, NU, MD, MC and SE to effectively raise the border minimum utility thresholds and further prunethe search space. On the other hand, TKO is the first one-phase algorithm developed for top-k HUI mining, which integrates the novel strategies RUC, RUZ and EPB to greatly improve its performance. Empirical evaluations on different types of real and synthetic datasets show that the proposed algorithms have good scalability on large datasets and the performance of the proposed algorithms is close to the optimal case of the state-of-threat two-phase and one-phase utility mining algorithms.

10. FUTURE WORK

In future, we will work on recommendation and we will proposed a new framework for top-k HUI mining, it has not yet been incorporated with other utility mining tasks to discover different types of top-k high utility patterns such as top-k high utility episodes, top-k closed high utility item sets, top-k high utility web access patterns and top-k mobile high utility sequential patterns. These leave wide rooms for exploration as future work.

REFERENCES

[1] R. Agrawal and R. Srikant, “Fast algorithms for mining associationrules,” in Proc. Int. Conf. Very Large Data Bases, 1994, pp. 487–499.
[2] C. Ahmed, S. Tanbeer, B. Jeong, and Y. Lee, “Efficient tree structuresfor high-utility pattern mining in incremental databases,”IEEE Trans. Knowl. Data Eng., vol. 21, no. 12, pp. 1708–1721, Dec.2009.
[3] K. Chuang, J. Huang, and M. Chen, “Mining top-k frequent patternsin the presence of the memory constraint,” VLDB J., vol. 17,pp. 1321–1344, 2008.
[4] R. Chan, Q. Yang, and Y. Shen, “Mining high-utility item sets,” inProc. IEEE Int. Conf. Data Mining, 2003, pp. 19–26.
[5] P. Fournier-Viger and V. S. Tseng, “Mining top-k sequentialrules,” in Proc. Int. Conf. Adv. Data Mining Appl., 2011, pp. 180–194.

[6] P. Fournier-Viger, C.Wu, and V. S. Tseng, “Mining top-k associationrules,” in Proc. Int. Conf. Can. Conf. Adv. Artif.Intell., 2012, pp. 61–73.
[7] P. Fournier-Viger, C. Wu, and V. S. Tseng, “Novel concise representationsof high utility item sets using generator patterns,” inProc. Int. Conf. Adv. Data Mining Appl. Lecture Notes Comput. Sci.,2014, vol. 8933, pp. 30–43.
[8] J. Han, J. Pei, and Y. Yin, “Mining frequent patterns without candidategeneration,” in Proc. ACM SIGMOD Int. Conf. Manag. Data,2000, pp. 1–12.
[9] J. Han, J. Wang, Y. Lu, and P. Tzvetkov, “Mining top-k frequentclosed patterns without minimum support,” in Proc. IEEE Int.Conf. Data Mining, 2002, pp. 211–218.
[10] S. Krishnamurthy, “Pruning strategies for mining high utilityitem sets,” Expert Syst. Appl., vol. 42, no. 5, pp. 2371–2381, 2015.
[11] C. Lin, T. Hong, G. Lan, J. Wong, and W. Lin, “Efficient updatingof discovered high-utility item sets for transaction deletion indynamic databases,” Adv. Eng. Informat., vol. 29, no. 1, pp. 16–27,2015.
[12] G. Lan, T. Hong, V. S. Tseng, and S. Wang, “Applying the maximumutility measure in high utility sequential pattern mining,”Expert Syst. Appl., vol. 41, no. 11, pp. 5071–5081, 2014.
[13] Y. Liu, W. Liao, and A. Choudhary, “A fast high utility item setsmining algorithm,” in Proc. Utility-Based Data Mining Workshop,2005, pp. 90–99.
[14] M. Liu and J. Qu, “Mining high utility item sets without candidategeneration,” in Proc. ACM Int. Conf. Inf. Knowl. Manag., 2012,pp. 55–64.
[15] J. Liu, K. Wang, and B. Fung, “Direct discovery of high utilityitem sets without candidate generation,” in Proc. IEEE Int. Conf.Data Mining, 2012, pp. 984–989.
[16] Y. Lin, C. Wu, and V. S. Tseng, “Mining high utility item sets in bigdata,” in Proc. Int. Conf. Pacific-Asia Conf. Knowl. Discovery DataMining, 2015, pp. 649–661.
[17] Y. Li, J. Yeh, and C. Chang, “Isolated items discarding strategy fordiscovering high-utility item sets,” Data Knowl. Eng., vol. 64, no. 1,pp. 198–217, 2008.
[18] G. Pyun and U. Yun, “Mining top-k frequent patterns with combinationreducing techniques,” Appl. Intell., vol. 41, no. 1, pp. 76–98,2014.
[19] T. Quang, S. Oyanagi, and K. Yamazaki, “ExMiner: An efficientalgorithm for mining top-k frequent patterns,” in Proc. Int. Conf.Adv. Data Mining Appl., 2006, pp. 436 – 447.
[20] Miss. A. A. Bhosale , S. V. Patil, Miss. P. M. Tare, Miss. P. S. Kadam“High Utility Item sets Mining on Incremental Transactions using UP-Growth and UP-Growth+ Algorithm”.
[21] Komal Surawase, Madhav Ingle “Mining Highly Utilized Item Set from Transaction Database”.
[22] Liang Wang, David Wai-Lok Cheung, Reynold Cheng, Member, IEEE, Sau Dan Lee, and Xuan S. Yang, “Efficient Mining of Frequent Item Sets on Large Uncertain Databases”, IEEE Transact ions On Knowledge And Data Engineering, volume 24, Issue No. 12, pp 2170-2183, DECEMBER 2012