ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 6, June 2018

# Mining Algorithm to Archive Top-K High Utility Itemset Using TKO with TKU

Reshma charbhe, Prof T. J. Parvat

Department of Computer Engineering, SIT Lonavala, Savitribai Phule Pune University, Pune, Maharashtra, India

**ABSTRACT:** Data mining is a computerized process of searching for models in large data sets that involve methods at the intersection of the database system. The popular problem of data mining is the extraction of high utility element sets (HUI) or, more generally, the extraction of public services (UI). The problem of HUI (set of elements of high utility) is mainly the introduction to the set of frequent elements. Frequent pattern mining is a widespread problem in data mining, which involves searching for frequent patterns in transaction databases. Solve the problem of the set of high utility elements (HUI) with some particular data and the state of the art of the algorithms. To store the HUI (set of high utility elements) many popular algorithms have been proposed for this problem, such as "Apriori", FP growth, etc., but now the most popular TKO algorithms (extraction of utility element sets) K in one phase) and TKU (extraction of elements sets Top-K Utility) here TKO is Top K in one phase and TKU is Top K in utility. In this paper, address previous issues by proposing a new frame work for k upper HUI where k is the desired number of HUI to extract. Extraction of high utility element sets is an uncommon term. But we are using it while shopping online, etc. It is part of the business analysis. The main area of application is the analysis of the market basket, where when the customer buys the item he can buy another to maximize the benefit both the customer and supplier profit.

**KEYWORDS**: utility mining, high utility itemset, top k- pattern mining, top- k high utility itemset mining.

## I. INTRODUCTION

There is a huge amount of data available in the information industry. Data mining is the efficient discovery of valuable and vivid information from a vast collection of data. Frequent set mining set (FIM) discovers the only frequent elements, but the set of HUI High Utility items. In the FIM profile of the set of elements are not considered. This is because the amount of the purchase does not take into account. Data mining is the process of analyzing data from different points of view and summarizing it in useful data. Data mining is a tool for analyzing data. It allows users to analyze data from different levels or angles, organize them and find the relationships between the data. Data mining is the process of finding patterns between enough fields in the large relational database. A classic algorithm based on Top K models consists of two phases. In the first phase, called phase I, it is the complete set of high transaction weighted utility itemset (HTWUI). In the second phase, called phase II, all HUIs are obtained by calculating the exact HTWUI utilities with a database scan. Although many studies have been devoted to the extraction of HUI, it is difficult for users to effectively choose an appropriate minimum threshold. Depending on the threshold, the size of the output can be very small or very large. Also the choice of the threshold significantly impacts the performance of the algorithms if the threshold is too low then too many HUI will be presented to users then it will be difficult for users to understand the results. A large amount of HUI creates data mining algorithms unproductive or out of memory, subsequently the more HUIs create the algorithms, the more resources they consume. Conversely, if the threshold is too high, HUI will not be found.

## II. RELATED WORK

Vincent S. Tseng, Bai-En Shie, Cheng-Wei Wu, and Philip S. Yu, Fellow Mining high utility thing sets from a regard based database intimates the disclosure of thing sets with high utility like points of interest. In spite of the fact that various applicable calculations have been proposed as of late, they cause the issue of creating countless thing sets for high utility things and so forth [1]. Such countless thing sets corrupts the mining execution as far as execution time and

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 6, June 2018

space necessity. The condition may end up being all the more horrendous when the database contains heaps of long trades or long high utility thing sets. In this paper, we propose two calculations, to be specific utility example development (UP-Growth) and UP-Growth+, for mining high utility thing sets with an arrangement of successful methodologies for pruning hopeful thing sets. The data of high utility thing sets is kept up in a tree-based information structure named utility example tree (UP-Tree) to such an extent that applicant thing sets can be created effectively with just two sweeps of database. The execution of UP-Growth and UP-Growth+ is differentiated and the best in class figuring's on various sorts of both honest to goodness and designed instructive accumulations. Trial comes to fruition show that the proposed estimations, especially UP-Growth+, decrease the amount of candidates enough and in addition beat diverse counts significantly to the extent runtime, especially when databases contain stores of long trades.

ChowdhuryFarhan Ahmed, Syed KhairuzzamanTanbeer, Byeong-SooJeong and Young-Koo Lee have introduced three new tree structures to effectively perform incremental and intuitive HUP mining [2]. The primary tree structure is used to orchestrate things as indicated by your lexicographical request. It is known as incremental lexicographic tree HUP (IHUPLTree). Incremental information can be acquired without reconstruction. The following tree structure is the IHUP Transaction Frequency Tree (IHUPTF Tree), which orchestrates things based on their exchange periodicity in the slip request. To reduce extraction time, the last tree is composed, IHUP-Transaction Weighted Utilization Tree (IHUPTWUTree). The structure of this tree depends on the TWU estimate of things in the scroll request.

Alva Erwin, Raj P. Gopalan and N. R. Achuthan, have proposed the calculation of the CTU-PROL for the efficient extraction of useful sets of things from expansive data sets [3]. This calculation finds the TWU expansions in the exchange database. In case the information collections are too large to be stored in the main memory, the calculation makes the subdivisions use parallel projections and for each subdivision, a tree of compressed utility schemes (CUP tree) is used to exploit all the high utility disposition establishes. In the remote possibility that the data set is small, it becomes a solitary CUP tree to extract very useful sets of things.

Shankar S., Purusothaman T., Jayanthi, S., recommended another calculation for extricating exceedingly helpful arrangements of components [4]. This quick utility extraction calculation (FUM) discovers every one of the arrangements of high utility components inside the imperatives limit of the utility gave. The proposed FUM calculation scales well as the measure of the exchange database increments concerning the quantity of unmistakable things accessible.

R. Chan, Q. Yang and Y. Shen have suggested that we extract very useful sets of elements [5]. They proposed a new mining idea aimed at higher K objectives, which focuses on closed top utility closed schemes. They add the concept of utility to capture highly desirable statistical models and present a level-wise element extraction algorithm. . They develop a new utility-based pruning strategy that allows pruning of low utility elements through a weaker monotonic condition of the ant hormones.

Ramaraju C., Savarimuthu N., proposed a new algorithm based on conditional trees for the creation of high utility objects [6]. A new conditional high availability tree (CHUT) compresses two-phase transactional databases to reduce the search space and a new algorithm called HU-Mine is proposed to extract a complete set of highly useful sets of elements.

Y. Liu, W. Liao and A. Choudhary proposed a rapid extraction algorithm for extremely useful sets of elements [7]. There is a two-step algorithm for efficiently reducing the number of candidates and accurately obtaining the complete set of high-use element sets. In the first phase, they propose a model that applies the "ownership of transaction-weighted closure" in the search space to accelerate the identification of candidates. The last phase identifies the highly useful set of elements.

Adinarayanareddy B., O. SrinivasaRao, MHM Krishna Prasad, suggested an improvement in the extraction of objects with high utility UP-Growth [8]. The compact tree, the utility model tree, which is UP-Tree, preserves the information of the transactions and their element sets. Facilitates mining performance and frequently avoids scanning the original database. UP-Tree analyzes the database only twice to obtain candidate elements and manage them in an efficient and structured way. UP-Growth requires more execution time for the second phase using UP-Tree. Therefore, they present modified algorithms with the aim of reducing execution times by effectively identifying highly useful sets of elements.

P. Asha, Dr. T. Jebarajan, G. Saranya, and presents a review of incremental productive calculation to extract 0useful sets of assets in a dynamic and transported database [9]. The proposed structure uses an ace center and two slave centers. The database is distributed for each slave concentrator for the calculation. The slave Hub corresponds to the event of each thing. This information is stored in your nearby table. At that point, each slave concentrator sends these tables to the center of the ace. Main node keeps the world table to store this information. In view of the estimate of the basic utility limit, it appears as promising and unpromising.

## III. MATHEMATICAL MODULES

1. S=Tp, Tu, TWU, RTU, Up tree, Up growth+, PHUI
2. S=system
3. Tp=Set of transaction with profit of each item
4. Tu=Transaction weighted utility
5. TWU= Transaction Weighted utility
6. RTU=Recognized Transaction utility
7. UUI = utility of Unpromising Item
8. UP tree= utility Pattern Tree.
9. UP growth+ = utility Pattern Growth +
10. Improved UP growth (Up growth +)= Advanced Utility Pattern growth
11. PHUI= Potentially High Utility Itemset
12. Task: Discovery of High Utility Itemset
13. Input : Data Base DB set of Itemset
14. Transaction T Belong DB i=1.k=1
15. Ip Internal Utility value of item
16. H: high Utility itemset
17. CK: Candidate S itemset
18. Minimum Utility value threshold minUtil
19. Output : High Utility Itemset H

**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 itemset based itemset (base –itemset 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
4. //Step 1:Calculate utility information of the node Q
5. Float twu=0,BU=0,SU=0,NU=0;

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 6, June 2018

```
6.      For each header table H in Hs do
7.      For each node N for the item Q in the tree T corresponding to H do
8.      BU+=T.N.bu;
9.      SU+=T.N.su;
10.     NU+=T.N.nu;// N.nu is a utility for item Q in the list N.piu
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
19.     // Step 3:Pass add-information on node Q to parent node
20.     Move each node's bac-info to its parent;
21.     End For
22.     Delete itemset whose value are less than minUti from TKHUIs;
23.     Return TKHULS;
24.     END
```

## IV. THE PROPOSED SYSTEM

In the proposed framework, we address the problems mentioned above by proposing another system for calculating the means and means responsible for a high utility configured in parallel extraction using TKU and TKO. Two types of production calculations called TKU (extraction of sets of utility elements Top-K) and TKO (sets of themes of extraction Top-K are proposed in one phase) to extract these series of elements without the need to establish a utility minimum. But the TKO algorithm have the main disadvantage of not mainly accumulating the result of TKO given the value of the garbage in the set of high utility items isthe result of the TKU algorithm is increased but the execution time is high, so the alternative solution is to find the efficient algorithm in the proposed combination of the TKO and TKU algorithm system. It can be said that the result of TKO Top K in one phase is given at the entrance of TKU Top K in the utility result of TKO and TKU is increased and the execution time is low. In the proposed system, a new algorithm is generated for combining the name TKO and TKU as TKO WITH TKU or TKMHUI Top k Main set of utility elements.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 6, June 2018

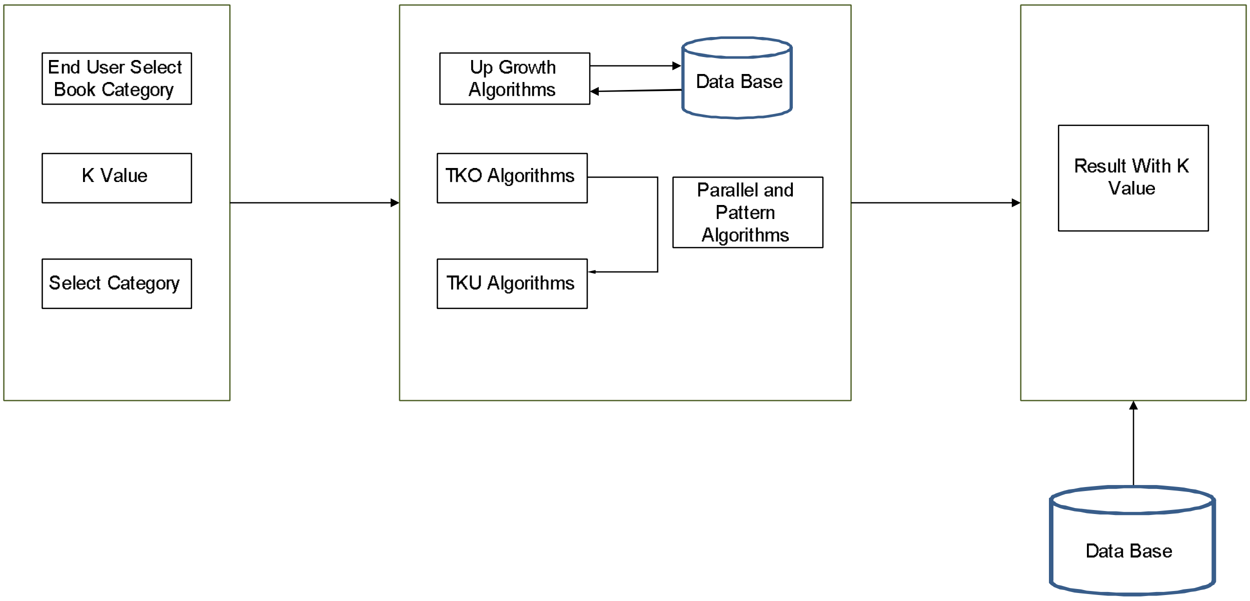



Fig.1 System Architecture

**Module:**

**Module 1 - Administrator (Admin)**

The administrator preserve database of the transactions made by customers. In the daily market basis, each day a new product is let go, so that the administrator would add the product or items, and update the new product view the stock details.

**Module 2 - User (Customer)**

Customer can purchase the number of items. All the purchased items history is stored in the transaction database.

**Module 3 - Construction of Up Tree**

In Up Tree Dynamic Table is generated by algorithms. Mainly the Up growth is considerable to get the PHUI itemset.

**Module 4 - TKO and TKU Algorithms**

In Combination of TKO and TKU algorithms first the TKO (Top k in one phase) algorithms is called and then output of TKO is given as the input of TKU (Top k in utility phases) then the actual result is TKU Result.

## V. SYSTEM ANALYSIS

As a result, Top K Algorithm is applied in different data sets, such as data sets, such as mushrooms, chess and accidents with different k, respectively. In this graph TKUWITHTKO is the best performance among HUI top-k mining algorithms, in this TKO chart with TKU Spend 120 almost time-based for seconds to complete the mining process while REPT and TKU last longer than 160 seconds and TKO 70 seconds. In comparison, the TKO and TKU algorithms of the table are compared with various parameters, such as TWU and CHUD, and the Garbage Find values.

In below Table 1 as per the observations of the current status of project is given by the comparison between the Algorithms TKO, TKU and TKO with TKU with following parameter such as the Anti-monotonicity, fallow TUU high utility pattern, Closed high utility data set and the Most important point Garbage value generator and Minimum Utility Value.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

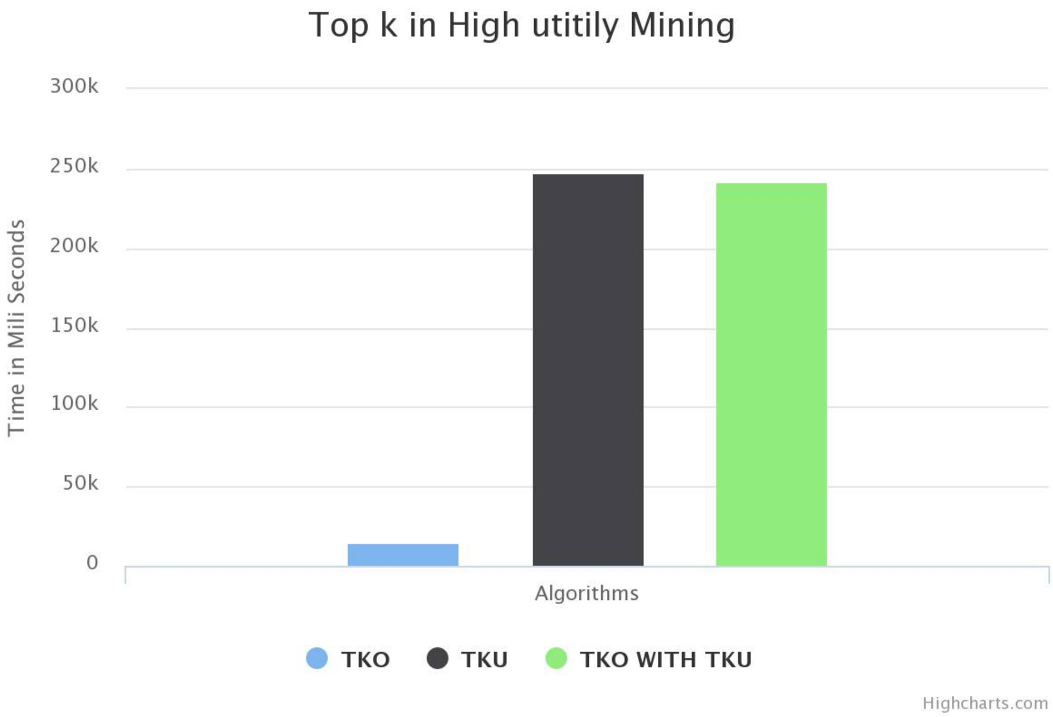
Vol. 6, Issue 6, June 2018

| Sr. No. | PARAMETRE | TKO | TKU | TKO With TKU |
|---|---|---|---|---|
| 01 | Anti-monotonicity | Yes | No | Yes |
| 02 | Fallow TWU high utility pattern | No | Yes | Yes |
| 03 | CHUD (Closet high utility dataset) | Yes | Yes | Yes |
| 04 | Finding Garbage Value in algorithm | Yes | No | No |
| 05 | Min until value | ==0 | ==0 | Set by Data set. |

Table1: Comparison between Algorithms

This graph shows that comparison between algorithms:

In this graph, the execution time of algorithms in milliseconds which are given in below table.




Graph 1: Comparison between Algorithms

| Number | Name of Algorithms | Time In MS | K value |
|---|---|---|---|
| 1 | TKO | 15000 | 3 |
| 2 | TKU | 247825 | 3 |
| 3 | TKO with TKU | 241426 | 3 |

This graph shows that comparison between Existing Algorithms TKO and TKU:

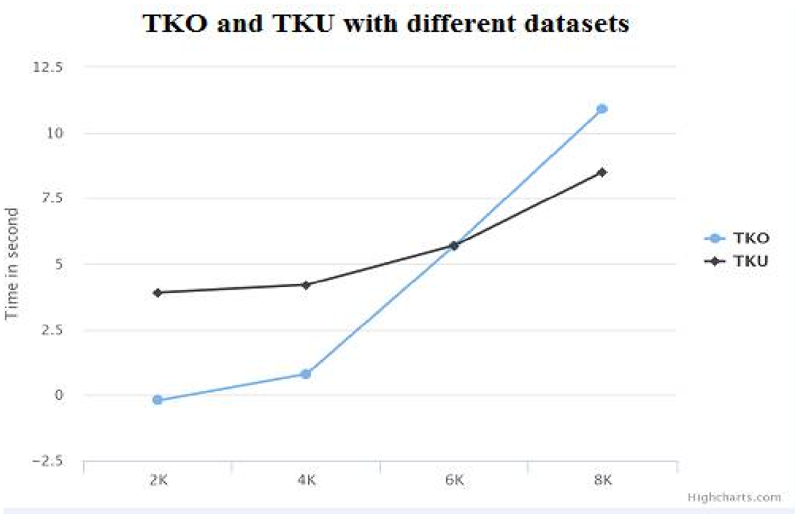
ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

**International Journal of Innovative Research in Computer and Communication Engineering**

*(A High Impact Factor, Monthly, Peer Reviewed Journal)*

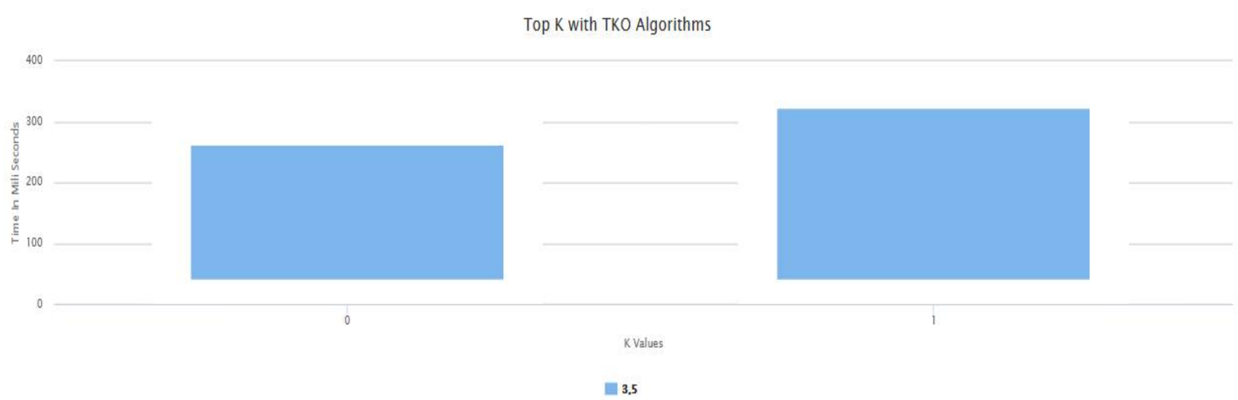
*Website: www.ijircce.com*

**Vol. 6, Issue 6, June 2018**



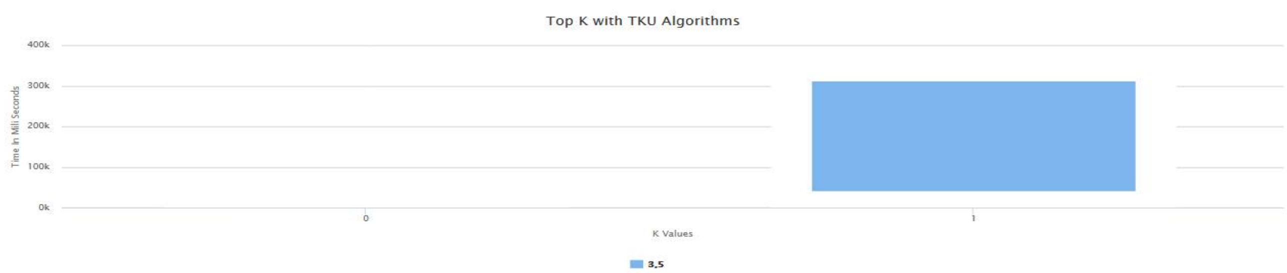

Graph 2: Comparison between Existing Algorithms TKO and TKU

This graph shows the different K value VS Time of TKO Algorithm:




Graph 3: Graph of Different K Value VS Time of TKO Algorithm

This graph shows the different K value VS Time of TKU Algorithm:




Graph 4: Graph of Different K Value VS Time with TKU Algorithm

This graph shows the different K value VS Time of TKO WITH TKU Algorithm:

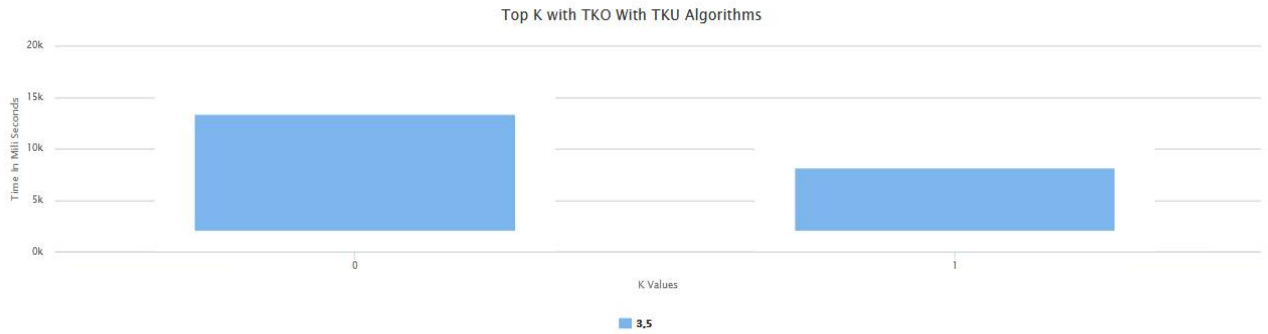
ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

**International Journal of Innovative Research in Computer and Communication Engineering**

*(A High Impact Factor, Monthly, Peer Reviewed Journal)*

Website: www.ijircce.com

**Vol. 6, Issue 6, June 2018**




Graph 5: Graph of Different K Value VS Time with TKO WITH TKU Algorithm

## VI. CONCLUSION

In this paper, we looked at the question of the best sets of high-use mining mines, where k is the coveted number of highly useful sets of things to extract. The most competent combination of TKO WITH TKU of the TKO and TKU calculations is proposed to extract such sets of objects without establishing utility limits. Instead TKO is the first single phase algorithm developed for top-k HUI mining called PHUI (high potential set of utility elements) and PHUI is given to TKU in the utility phases. Empirical evaluations on different types of real and synthetic data sets display the proposed algorithms have good scalability in large data sets and the performance of the proposed algorithms are close to the optimal case of the state of the combination of both phases in an algorithm.

## REFERENCES

1. Vincent S. Tseng, Bai-En Shie, Cheng-Wei Wu, and Philip S. Yu, Fellow, IEEE "Efficient Algorithms for Mining High Utility Itemsets from Transactional Databases" IEEE Trans. Knowledge and Data Engineering, vol. 25, no. 8, August 2013.
2. ChowdhuryFarhan Ahmed, Syed KhairuzzamanTanbeer, Byeong-SooJeong, and Young-Koo Lee, Member, IEEE "Efficient Tree Structures for High Utility Pattern Mining in Incremental Databases" IEEE Trans. Knowledge and Data Engineering, Vol. 21, Issue 12, pp. 1708-1721, December 2009.
3. Alva Erwin, Raj P. Gopalan, and N. R. Achuthan, "Efficient Mining of High Utility Itemsets from Large Datasets", In Proc. of PAKDD 2008.
4. Shankar, S.; Purusothaman, T.; Jayanthi, S. "Novel algorithm for mining high utility itemsets" International Conference on Computing, Communication and Networking, Dec. 2008.
5. Raymond Chan; Qiang Yang; Yi-Dong Shen, "Mining high utility itemsets" In Proc. of Third IEEE Int'l Conf. on Data Mining ,November 2003.
6. Ramaraju, C., Savarimuthu N. "A conditional tree based novel algorithm for high utility itemset mining", International Conference on Data mining, June 2011.
7. Ying Liu, Wei-keng Liao, AlokChoudhary "A Fast High Utility Itemsets Mining Algorithm" In Proc. of the Utility-Based Data Mining Workshop, 2005.
8. Adinarayanareddy B ,O SrinivasaRao, MHM Krishna Prasad, "An Improved UP-Growth High Utility Itemset Mining" International Journal of Computer Applications (0975-8887) Volume 58-No.2, November 2012
9. P. Asha, Dr. T. Jebarajan, G. Saranya, "A Survey on Efficient Incremental Algorithm for Mining High Utility Itemsets in Distributed and Dynamic Database" IJETAE Journal, Vol.4, Issue 1, January 2014.