

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 7, July 2017

Design of Search Engine Considering Top k High Utility Item set (HUI) Mining

Sanjana S. Shirsat, Prof. S. A. Joshi

Department of Computer Network, Sinhgad College of Engineering, Pune, Savitribai Phule Pune University
Maharashtra, India

Department of Computer Engineering, Sinhgad College of Engineering, Pune, Savitribai Phule Pune University
Maharashtra, India

ABSTRACT: Frequent Item sets discovery of sets of items that are frequently purchased together by customer. High Utility Item sets (HUIs) refers to discovery of an item set with high utility such as streaming analysis, market analysis, mobile computing and biomedicine. HUIs consider with particular attribute of the item set. HUIs mining is defined as qualitative representation of user preferences. The primary goal of mining is to discover hidden patterns and unexpected trends in data set with HUI and the secondary goal is to discover Top-k results. Point of Interest (POI) is to provide personalized recommendation of particular aspects such as restaurants, movie theaters. Recommendation is depends upon user's previous interest or previous search options.

HUIs mining identifies item sets where its attribute satisfies threshold. According to the threshold, items categorized as High Utility Item sets and Low Utility item sets. Resulting candidate item sets further undergoes pruning techniques so as to reduce no. of candidate item set considering. Efficient framework can be designed for mining Top-k HUIs set where k is no. of HUIs to be mined and defining the recommendation and POI.

HUI mining is used to find Top-k results to enhance decision making process. HUIs mining is useful for building powerful search engines to display Top-k results. IT has many application like market analysis, stream analysis and biometric.

KEYWORDS: Utility mining, high utility item sets, high utility item set mining, top-k, pattern mining, parallel mining, top-k high utility item set mining, Data mining ,frequent item set, transactional database.

I. INTRODUCTION

A search engine is an information retrieval system constructed to find information stored on a computer system. The search results are usually shows in a list and are commonly called hits. Search engines are helps to minimize the time required to find information and the amount of information which must be consulted. Search engines give an interface to a group of items that enables users to specify criteria about an item of interest and have the engine that find the matching items. The criteria are making reference to as a search query. In the case of text search engines, the search query is typically given as a set of words that identify the desired concept that one or more documents may contain. Data mining is use to find hits according to some conditions. Data mining is concerned with analysis of large volumes of data to automatically discover interesting regularities or relationships which in turn leads to better understanding of the underlying processes. The primary goal is to discover hidden patterns, unexpected trends in the data. Frequent item set mining (FIM) [1][9][10] is a fundamental research topic in data mining. However, the traditional FIM may invent 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 determine item sets with high utilities such as high profits. To address these issues, utility mining is 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 show 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) [2][7] if its utility is no less than a user-specified minimum utility threshold

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 7, July 2017

min-util. 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 Top k utility Itemsets (TKU) model is 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-util, 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 TKU[7][1] 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-util[1][7] 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. Another algorithm used is Top K utility Item set in One phase(TKO)[7]. This algorithm can discover top k HUI in one phase.

Web page:

Search engine is information retrieval system in which the search results are usually presented in a list. Search engine is considering the data mining to find out the appropriate information. In existing application or web page it takes the frequent item sets mining which is depend upon frequency of the item sets. Frequent item set mining does not consider attribute. For consideration of attribute uses HUI mining. Introducing a book web page consist of search engine which gives top-k high utility item sets.

In this k no of item sets are not fixed. Following are contribution in book web page:

- High Utility Item sets: HUI generated by considering particular attribute. For book web page defining attribute in terms of hit and buy. Utility of the item set is decided by hit and buy. Secondly considering attribute as a price.
- Top-k results: providing k no of results to the user. K value is not defined.
- Recommendation: in shopping application Recommendation is based on user history but which is store in browsers cookie. If those cookies are deleted then recommendation will not get generate. Recommendation is depends upon user previous interest or previous search options.
- Point of interest (POI): Point of Item sets (POI) is to provide personalized recommendation of particular aspects such as restaurants, movie theaters. According to user interest suggesting some more products which are related.

II. REVIEW OF LITERATURE

Hua-Fu Li, Hsin- Yun Huang et. al. Fast memory efficient mining of high utility item sets in data streams [2], this paper proposed two efficient one pass algorithms, MHUI-BIT and MHUI-TID for mining high utility item set from data streams within a transaction sensitive sliding window. Two effective representations of item information and an extended lexicographical tree based summary data structure are developed to improve the efficiency of mining high utility item sets.

T.Vinothini, V.V.Ramya Shree Mining High Utility Item sets from Large Transactions using Efficient Tree Structure [4] clarifies; mining high utility item sets from a transactional database refers to the discovery of item sets with high utility like profits. It is an expansion of the frequent pattern mining. Although a number of relevant algorithms have been proposed in recent years, they incur the problem of producing a large number of candidate item sets for high utility item sets. Such a 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

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 7, July 2017

contains lots of long transactions or long high utility item sets. Association rule mining with frequent item sets are used to extract item set relationships. Frequent pattern mining algorithms are constructed to find commonly occurring sets in databases. Memory and run time requirements are high in frequent item set pattern mining algorithms. Systolic tree structure is a reconfigurable architecture used for utility pattern mining operations. High throughput and faster execution are the features of the systolic tree based reconfigurable architecture. The systolic tree mechanism is used in the frequent pattern extraction process for the transaction database. Systolic tree based rule mining scheme is enhanced for Weighted Association Rule Mining (WARM) process, used to fetch the frequently accessed transaction database with its weight values. The dynamic dataset is weight assignment scheme uses the item request count and span time values. The proposed system improves the weight estimation process with span time, request count and access sequence details.

J. Yin, Z. Zheng, L. Cao, Y. Song, and W. Wei, Mining top-k high utility sequential patterns[5] clarifies, a novel framework called top-k high utility sequential pattern mining to tackle this critical problem. Accordingly, an efficient algorithm, Top-k high Utility Sequence (TUS for short) mining, is created to identify top-k high utility sequential patterns without minimum utility. In addition, three effective features are received to handle the efficiency problem, including two strategies for raising the threshold and one pruning for filtering unpromising items. Experiments are organized on both synthetic and real datasets.

Vincent S. Tseng, Bai-En Shie, Cheng-Wei Wu, and Philip S. Yu, Fellow, Efficient Algorithms for Mining High Utility Item sets from Transactional Databases[5] clarifies, mining high utility item sets from a transactional database refers to the discovery of item sets with high utility like profits. Although a number of relevant algorithms have been proposed in recent years, they incur the problem of producing a large number of candidate item sets for high utility item sets. Such a 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 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. The performance of UP-Growth and UP-Growth+ is compared with the state-of-the-art algorithms on many types of both real and synthetic data sets. Experimental results show that the proposed algorithms, especially UP Growth+, not only reduce the number of candidates effectively but also outperform other algorithms substantially in terms of runtime, especially when databases contain lots of long transactions.

Cheng-Wei Wu, Fournier-Viger, Philip S. Yu, Vincent S. Tseng Efficient Algorithms for Mining the Concise and Lossless Representation of Closed+ High Utility Item sets [3] clarifies, mining high utility itemsets (HUIs) from databases is an important data mining task, which refers to the discovery of itemsets with high utilities (e.g. high profits). However, it may present too many HUIs to users, which also degrades the efficiency of the mining process. To achieve high efficiency for the mining task and provide a concise mining result to users, a novel framework in this paper for mining closed+ high utility itemsets (CHUIs), which serves as a compact and lossless representation of HUIs. 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 algorithms are very efficient and that our approaches achieve a massive reduction in the number of HUIs.

R.Nandhini, Dr.N.Suguna Shrewd Technique for Mining High Utility Item set via TKU and TKO Algorithm[4] clarifies, High utility item sets (HUIs) mining is an emanate topic in data mining, which refers to discovering all item sets having a utility meeting a user-specified minimum utility threshold. Used three efficient algorithm ApriorCH (Apriori-based algorithm for mining High utility closed item set), Apriori HC-D (AprioriHC algorithm with removing unpromising and isolated items) and CHUD (Closed High Utility item set Discovery) algorithm. However, setting min-util appropriately is a hard problem for users. Finding an appropriate minimum utility threshold by trial and error is a tough process for users. Setting very low min-util, HUIs will be generated in large, which may cause the mining process to be very inefficient. In another way, by setting too high min-util, no HUI will be found. So the above issues is addressed by proposing a framework for mining top-k high utility item set , where k is the desired number of HUIs to

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 7, July 2017

be mined. Two types of proficient algorithms named 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 the need to set min-util.

Vincent S. Tseng, Senior Member, IEEE, Cheng-Wei Wu, Philippe Fournier-Viger, and Philip S. Yu, Efficient Algorithms for Mining Top-K High Utility Item sets[7] clarifies, High utility item sets (HUIs) mining is an rising topic in data mining, which refers to discovering all item sets having a utility meeting a user-specified minimum utility threshold min-util. However, setting min-util appropriately is a challenging problem for users. Finding an appropriate minimum utility threshold by trial and error is an exhausting process for users. If min-util is set too low, too many HUIs will be found which may cause the mining process to be very inefficient. On the other hand, if min-util is set too high, it is accepted that no HUIs will be found. In this paper, address the above issues by proposing a new scheme for top-k high utility item set mining, where k is the desired number of HUIs to be mined. Two types of efficient algorithms named TKU (mining Top-K Utility item sets) and TKO (mining Top-K utility item sets in One phase) are recommended for mining such item sets without the need to set min-util. This paper provides a structural comparison of the two algorithms with discussions on their advantages and limitations. Observational evaluations on both real and synthetic datasets show that the performance of the proposed algorithms is close to that of the optimal case of state-of-the-art utility mining algorithms.

III. EXISTING SYSTEM APPROACH

Frequent Item set mining is a data analysis method which originally developed for market based analysis and aims at finding regularities in shopping behavior. But frequent item set mining consider only frequency. For frequent item set mining mining algorithm used such as Aprior, FP growth etc. Frequent Item set only consider the frequency of the item does not consider the attribute. To overcome this problem HUI mining introduce which consider the attribute of the item sets. Basic method for generating HUI are Aprior CH(Aprior based algorithm for mining High Utility closed item set), Aprior HV-D(Aprior algorithm with removing unpromising itemset) and CHUD(Closed High Utility Item set Discovery algorithm) [1]. In these algorithm setting min-util appropriately is hard problem. In these process setting lower utility generate very inefficient results. these algorithms required more time. Another algorithm based on tree structure which are up growth and upgrowth+ [5]. These algorithm maintain the tree based data structure named Up tree. up growth and up-growth+ decreases the no of candidate and make the performance much better. Extension to up-growth+ algorithm TKU algorithm purposed mining without need to generate min-util . TKU [7] consist of two phases constructing up tree, generating Potential High Utility Item sets(PHUI) and identifying top k HUI from PHUI. TKO [7] is algorithm discover HUI in one phase. It utilize basic search procedure HUI miner and its utility list structure. TKO have best performance considering time.

IV. SYSTEM OVERVIEW

UP-growth[6] algorithm discovering high utility item sets are efficient. By applying the proposed strategies of these algorithms the number of generated candidate item sets can be highly decreased in phase I and high utility item sets can be identified more efficiently in phase II. This technique is useful on static data sets. It did not consider the adaption of database. Our proposed system will perform on transactional database which is related to data mining. i.e. deletion or insertion of one or more records from database will consider on database history. To achieve this it uses the existing approach. Proposed system can avert unnecessary or repetition of calculations by using previous results when a database is updated, or when the threshold value is replaced. UP-Growth Algorithm: The UP-Growth is one of the productive algorithms to generate high utility item sets depending on construction of a global UP Tree. In this system using UP Growth+ maintain the item set dataset by creating UP tree. TKU and TKO find out high utility item sets by both pattern mining and parallel mining. At first stage TKO is applied on the item set database with both parallel and pattern mining. In second step TKU is applied on the item set database with both parallel and pattern mining. The efficient Top-k high utility item sets are generated. Two efficient algorithms named TKU (mining Top-K Utility item sets) and TKO (mining Top-K utility item sets in one phase)[7] are proposed for mining the complete set of top-k HUIs in databases without the need to specify the min-util 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.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 7, July 2017

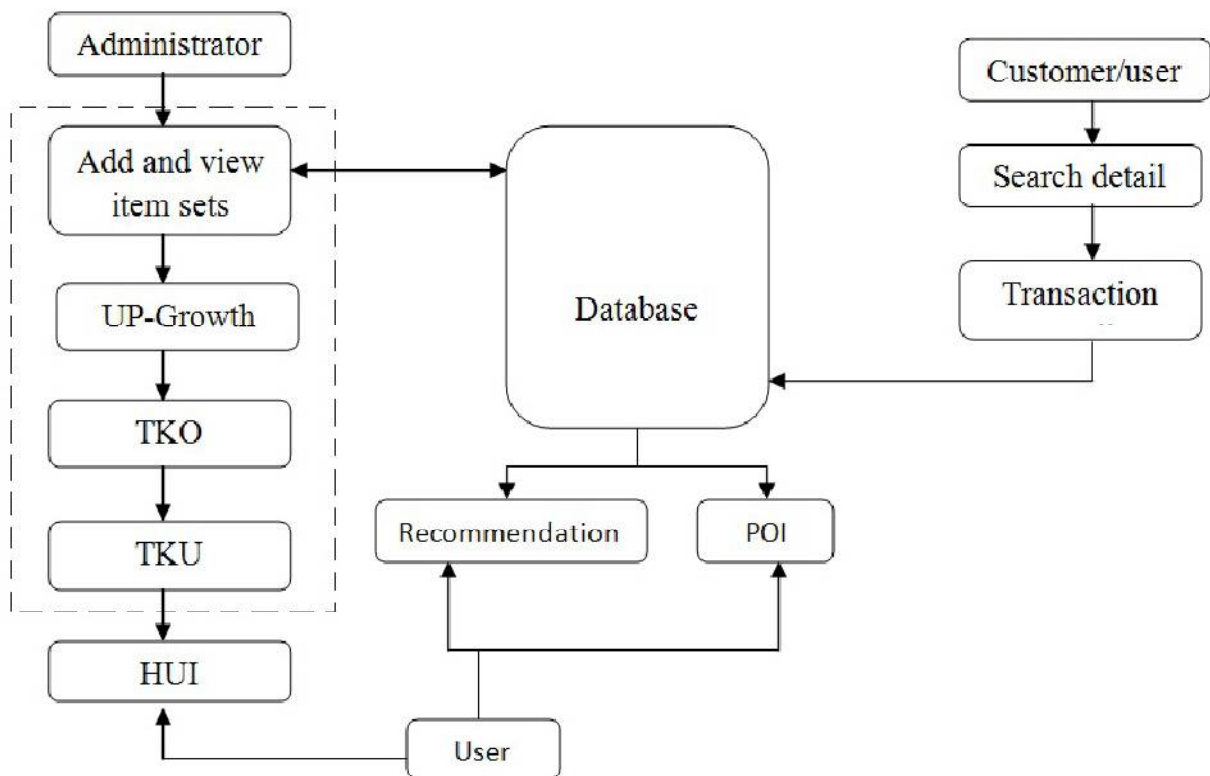


Fig. System Block Diagram

V. SYSTEM ANALYSIS

High Utility Item sets mining from transactional database constructed in 4 modules.

Module 1: Administrator

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

User must login the web page to access the page. After login category wise books will be displayed. User can purchase the number of items. All the purchased items history is stored in the transaction database.

Module 3: UP-Growth Algorithm

Up-Growth[6] forms the categorization of the data set and gives the categories tables. It measures the Maximum transaction Weighted Utilization (MTWU) from all items and considering multiple of min-util as a user specified threshold value.

Module 4: TKU and TKO algorithm

Top K Utility Item sets (TKU) and mining Top K Utility Item sets in One phase (TKO)[7] takes High utility mining on item set database. For mining it will consider both parallel and pattern mining.

Module 5: Recommendation and POI

Generate Recommendation and POI according to user history and interest.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 7, July 2017

VI. METHODOLOGY

- Up-growth algorithm applied to database to classifying all the categories. Input given to the UP growth+ will be the dataset of the items. Output expected to generate classification of the Data set items.
- TKO algorithm will applied to generate HUI. Classified database by UP Growth will be used as input to TKO. Output expected is to get Top k high utility itemsets with garbage values.
- On the results of TKO algorithm will apply TKU algorithm to give efficient results without garbage values in sufficient time. Output of TKO will be used as input to TKU. Output expected is to get Top k high utility itemsets.
- Defining the Recommendation and Point of Interest (POI) according to user interest.
- This system generate HUI by setting attribute.

Top K Utility Item sets in One phase (TKO) algorithm:

The next algorithm is Top K Utility Item sets in One phase (TKO) algorithm. In can discover top-k HUIs in one phase. It utilize basic search procedure of HUI-miner and it's utility list structure. But as in the results it generates garbage values.

Algorithm:

Input: 1) $u(P)$: Utility list for prefix P.

2) $Class[P]$: a set of item sets w. r. t. the prefix P.

3) $ULS[P]$: a set of utility lists w. r. t. P.

4) d : border minimum utility threshold $min_utility_{border}$.

5) TopK-CI-List: List of storing candidate item sets.

Output: Top k HUIs

- 1) For each $X = \{x_1, x_2, \dots, x_L\} \in Class[P]$ do
- 2) If ($\sum(X.iutils) \geq d$)
- 3) //Raise $min_utility_{border}$ by strategy RUC.
- 4) $d \leftarrow RUC(X, Topk-CI-List)$;
- 5) End if.
- 6) If ($(\sum(X.iutils) + \sum(X.rutils)) \geq d$)
- 7) $Class[X] \leftarrow \phi$, $ULS[X] \leftarrow \phi$.
- 8) For each $Y = \{y_1, y_2, \dots, y_L\} \in Class[P] \mid y_L \geq x_L$ do
- 9) $Z \leftarrow X \cup Y$;
- 10) $ul(Z) \leftarrow construct(ul(p), X, Y, ULS[P])$
- 11) $Class[x] \leftarrow Class[X] \cup Z$;
- 12) $ULS[X] \leftarrow ULS[X] \cup ul(Z)$;
- 13) End for.
- 14) Topk-HUI-Search($X, ULS[X], Class[X], d, TopK-CI_List$)
- 15) End if.
- 16) End for.

Top K Utility Item sets (TKU) algorithm:

This section describe Top K Utility Item sets (TKU) algorithm for discovering HUI with out specifying min-utility. TKU is an extension of a tree based up growth algorithm.\\

TKU algorithm execute in three steps:

- Constructing the UP tree.
- Generating Potential top K High Utility Item set (PKHUI).
- Identifying top HUI from PKHUI.

Input: 1) Database D;

2) The number of desired HUIs k;

3) Result by TKO;

Output: The complete set of PKHUIs C;

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 7, July 2017

- 1) Set $\min_utility_{border} \leftarrow 0$; TopK-MIU-List $\leftarrow \emptyset$; $C \leftarrow \phi$;
- 2) Construct up tree by scanning D twice;
- 3) // apply up-growth search procedure to generate PKHUIs;
- 4) For each PKHUI generated with estimated utility $ESTU(X)$ do
- 5) If $(ESTU(X) \geq \min_utility_{border} \text{ and } MAU(X) \geq \min_utility_{border})$
- 6) Output X and $\min\{ESTU(X), MAU(X)\}$; $C \leftarrow C \cup X$;
- 7) If $(MIU(X) \geq \min_utility_{border})$
- 8) //Raise min utility border.
- 9) End if.
- 10) End if.
- 11) End if.

VII. RESULT ANALYSIS

In results, it compares two methods TKO and TKU with proposed method.

Figure shows graph is draw between no. of item set in database and time means algorithms generated the time which means the total time taken to get the results of proposed system. X axis denotes the no. of item sets in the database and Y axis denotes the time require to operate the algorithm. TKU requires more time as compare to TKO but TKU gives more efficient results as compare to the TKO algorithm as shown in figure 8.11. By applying single TKO algorithm it gives correct result but in the garbage values. To overcome this problem proposed system apply TKU algorithm which gives more efficient result as compare to the separate implementation of TKO and TKU. Common feature of all three algorithms is time increases as number of item set in data set increases. Proposed approach gives efficient and effective result.

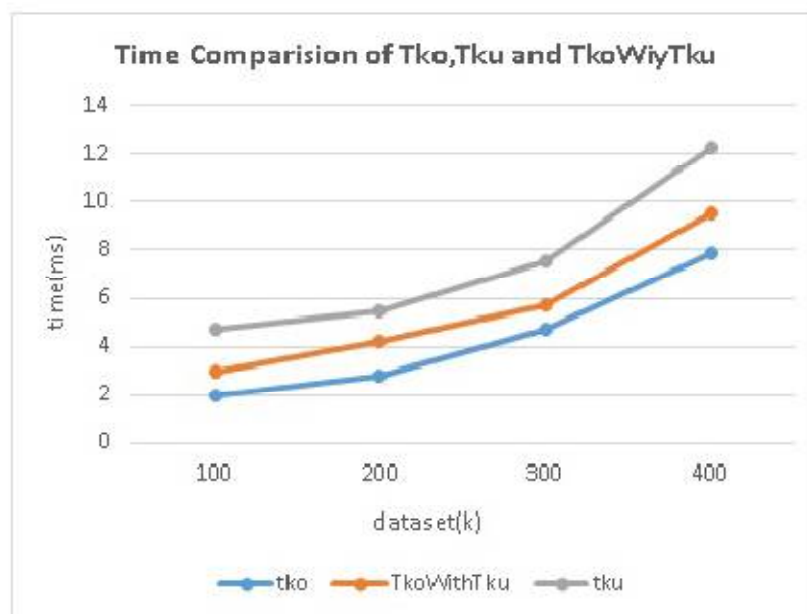


Fig. Result Analysis.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 7, July 2017

Figure shows the time graph which shows the time required for the system to generate top K HUI search. The input is search query for finding top K HUI. This graph represent time required to execute the query and get the result i.e top K HUI. This time is tested on available book database.

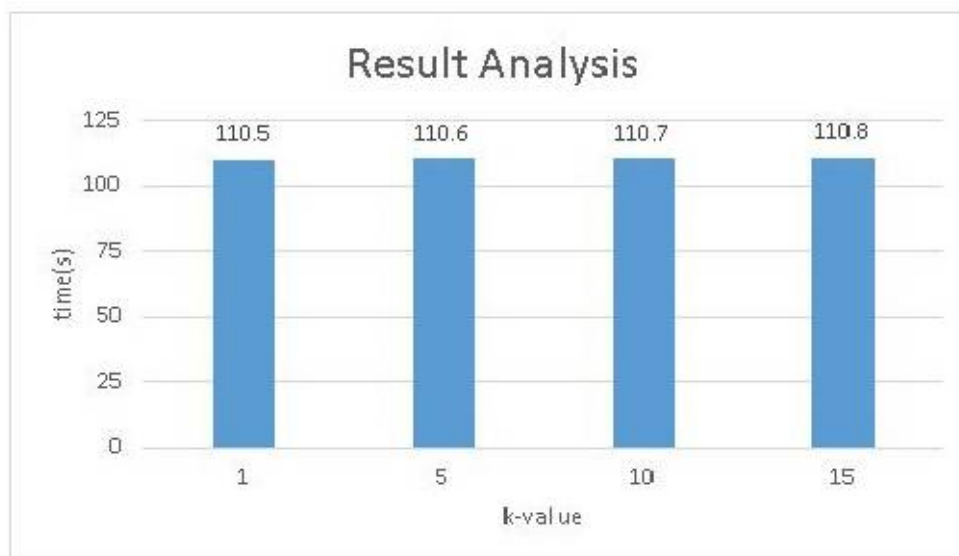


Fig. Time graph for top K HUI search

VII. CONCLUSION AND FEATURE WORK

The problem of top-k high utility item sets mining, where k is the desired number of high utility item sets to be mined is studied. 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. TKO is the one-phase algorithm developed for top-k HUI. TKU is the two-phase algorithm for mining Top-k high utility item sets. TKU effectively raise the border minimum utility thresholds and further prune the search space. Proposed system generate top k HUI by applying both TKO and TKU to get more efficient results with attribute hit and buy. TKO algorithm gives the garbage values so apply the TKU algorithm with more appropriate results in the sufficient time. Proposed system also generates recommendation and POI according to user history and user interest.

FUTURE WORK

- In future work system can be designed with both frequent item set mining and HUI mining.
- In future work deduplication can be applied on the database to check the duplications.

REFERENCES

- [1] R.Nandhini, Dr.N.Suguna "Shrewd Technique for Mining High Utility Item set via TKU and TKO Algorithm ", International Journal of Computer Science and Information Technologies, Vol. 6, 2015.
- [2] Hua-Fu Li, Hsin- Yun Huang et. al. "Fast memory efficient mining of high utility item sets in data streams", Eight international conference on Data Mining, 1550-4784, 2008.
- [3] Cheng-Wei Wu, Fournier-Viger, Philip S. Yu, Vincent S. Tseng "Efficient Algorithms for Mining the Concise and Lossless Representation of Closed+ High Utility Itemsets ", IEEE Trans. Knowl. Data Eng. 1041-4347, 2014.
- [4] T.Vinothini, V.V.Ramya Shree "Mining High Utility Item sets from Large Transactions using Efficient Tree Structure ", International Journal Of Latest Trends in Engineering and Technology(IJLTET), vol.6, issue, 2012.
- [5] J. Yin, Z. Zheng, L. Cao, Y. Song, and W. Wei, "Mining top-k high utility sequential patterns ", Data Mining (ICDM), IEEE 13th International Conference, 2013.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 7, July 2017

- [6] Vincent S. Tseng, Bai-En Shie, Cheng-Wei Wu, and Philip S. Yu, "Efficient Algorithms for Mining High Utility Itemsets from Transactional Databases", IEEE Trans. Knowl. Data Eng., VOL. 25, 2013.
- [7] V. S. Tseng, C. Wu, P. Fournier-Viger, and P. S. Yu, "Efficient algorithms for mining the concise and lossless representation of high utility itemsets," IEEE Trans. Knowl. Data Eng., vol. 28, no. 1, pp. 726739, Jan 2016.
- [8] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules", in Proc. Int. Conf. Very Large Data Bases, 1994, pp. 487-499.
- [9] Sudip Bhattacharya, Deepti Dubey, "High Utility Item set Mining", International Journal of Emerging Technology and Advanced Engineering, vol. 2, Issue 8, Aug 2012.
- [10] Junqiang Liu, Ke Wang, Benjamin C. M. Fung, "Mining High Utility pattern in one phase without generating candidates", IEEE Trans. Knowl. Data Eng. vol. 28, No. 5, May 2016.
- [11] Vincent S. Tseng, Chung-Jung Chu, Tyne Liang, "Efficient mining of Temporal High Utility Item set from Data streams", Aug 2006.
- [12] Hong Yao, Howard J. Hamilton, Liqiang Geng, "Unified Framework for Utility Based Measures for Mining Itemsets", Aug 2006.
- [13] Chowdhury Farhan Ahmed, Syed Khairuzzaman Tanbeer, Byeong-soo Jeong and Young-Koo Lee "Efficient tree structure High Utility Pattern mining in Incremental Database", IEEE Trans. Knowl. Data Eng., vol. 21, no. 12, Dec 2009.
- [14] J. Liu, K. Wang, and B. Fung, "Direct discovery of high utility itemsets without candidate generation", in Proc. IEEE Int. Conf. Data Mining, 2012, pp. 984-989.
- [15] U. Yun and J. Kim, "A fast perturbation algorithm using tree structure for privacy preserving utility mining", Expert Syst. Appl., vol. 42, no. 3, pp. 1149-1165, 2015.
- [16] C. Ahmed, S. Tanbeer, B. Jeong, and Y. Lee, "Efficient tree structures for high-utility pattern mining in incremental databases", IEEE Trans. Knowl. Data Eng., vol. 21, no. 12, pp. 1708-1721, Dec. 2009.