International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2018 IJSRCSEIT | Volume 3 | Issue 3 | ISSN : 2456-3307

Extraction of Top K Itemsets From High Utility Itemsets Using Faster High-Utility Itemset Miner

M. Geetha[1], S. Kavitha[2]

[1]Research Scholar, Department of Computer Science, Sakthi college of Arts and Science For Women, oddanchatram, Tamil Nadu, India

[2]Head and Associate Professor, Department of Computer Science,Sakthi college of Arts and Science For Women, oddanchatram, Tamil Nadu, India

ABSTRACT

Frequent itemset mining is the recent research topic in the data mining systems. It generally composes of tremendous volume of frequently searched/retrieved item with low/ high itemset values. This dilemma doesn't satisfy the user's requirements. The utility itemsets is an important topic and it can be measure in terms of weight, value, quantity and all other information's depending on the user's requirements. If the utility itemset is no less than user specified min utility, so this itemset is called a utility of high itemset. It contains a many applications like biomedicine, mobile computing, market analysis, etc. In database, the HUI is a difficult, because in FIM used the downward closer property is does not hold the utility of itemsets. Superset the low utility itemset can be a high utility so the3 HUI pruning search space is also difficult. To overcome this issue, we discovered fittest threshold for mining the relevant itemsets from set of itemsets. Setting of min-util value to the user is a daunting task. In order to find an efficient threshold value for the users, the behaviors of the users are studied. In this work, we proposed two mechanisms, namely, mining top k utility itemsets and mining top k utility itemsets in single phase in which k is the number of covered HUI mining. Initially, 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: Cloud computing, Cloud security, Peer to Peer, Resource Description Framework.

I. INTRODUCTION

Data mining is the field of our study. The applications of data mining are tremendously growing due to the growth of information technologies. In general context, data mining is explained as follows:

(i) Extracting the relevant knowledge from the set of unidentified or identified set of resources.
(ii) The formation of meaningful pattern by exploring the data in a hyperplane system.

The real world data may be in structured or unstructured form. The main objective of the work is to find the relations or similarity between the data for deriving useful knowledge. The behavior of data implies lot of information from its elementary form. It also plays a vital role in the data analysing process. It authorizes users to analyze data from several diverse dimensions or angles, categorize it, and sum up the relationships acknowledged.

Several users make use of data mining for discovering the knowledge from variant aspects. In some cases, knowledge discovery is a developmental step in the

data mining process. It comprises of several steps as follows:

- ✓ Cleaning the data which removes the noise or inaccurate data.
- ✓ Merging the data for achieving better data availability.
- ✓ Selecting the data for achieving better data retrievability.
- ✓ Transformation of data that completely migrates the data into better interpretability form.
- ✓ Mining the data that extracting the relevant data patterns.
- ✓ Evaluating the patterns that derive the data from the similar patterns.
- ✓ Presenting the knowledge from the derived databases.

1.1 Working procedure of data mining systems

The data analysis may be carried out in large or small scale data. Based on their data association, the data are classified or clustered. The queries are transacted over those datasets with unrestricted no. of queries. It has been applied to various fields like statistical, machine learning and neural networks. The data relationship is constructed from four models:

Classes: Initially, the received data is stored in an indexed form. The similar data is grouped based on their representative classes. By doing so, it reduces the effects of data traffic.

Clusters: Based on their logical relationships, the data are clustered.

Association: Based on the consecutive purchasing behavior of the users, the data are associated.

Sequential mining: In order to predict the behavior patterns, the past knowledge are used for finding the pattern similarity score. Depends on those score, the relevant data is achieved.

1.2 Characteristics of Data Mining

The main characteristics of the data mining are listed as follows:

Higher volume of data: It is enormous in real time process. Each data has to be analyzed effectively for constructing the relationships.

Incomplete of the data: It differentiates the quality of data from its original resources.

Data structure: It is complex in nature which predicts the statistical analysis process. The stored data may be in heterogeneous form

1.3 Benefits of Data Mining

The advantages of the data mining system are the:

- ✓ It is one of the best data rendering services.
- ✓ It depicts the better data retrieving services.
- ✓ It also helps to store and retrieve the data based on their behavior.
- ✓ It also genuinely derives the valuable information.
- ✓ It loads the data with the association of data system.
- ✓ It develops the better relationship with each other.
- ✓ It helps to develop extraordinary data promotion systems.

The stored data is flexible in nature.

1.4 Application of Data Mining

The main applications of the data mining are:

Marketing systems: In the marketing field, the data is constructed based on their historical data. It helps for promoting their brands via direct mail, online marketing, campaign etc. In order to maintain their retaining strategy, data mining techniques are widely adopted. As a result of market basket analysis, a store can include an appropriate production collection in a way that customers can buy frequent buying products in concert with satisfying. In addition it facilitates the retail companies to offer positive discounts for particular products by that it will pull towards a lot of customers.

Banking systems : To derive the knowledge from the financial data, data mining techniques are widely studied. Based on the historical transactional data, the loan prediction system is formed. Data mining aids banks identify fraudulent credit card

transactions to save from harm credit card's owner.

Manufacturing applications: In order to find any errors in the equipments, the parameter optimization developed using data mining process are employed. Still, some imperfection may occur due to the ranges of control parameters. Then, those optimal control parameters are utilized to manufacture wafers with preferred eminence.

Governments scenario: Data mining facilitates government agency by means of excavating and analyzing records of financial transaction to build patterns that can identify money decontaminate or criminal activities.

Law oriented application : Data mining can assists law enforcers in recognizing illegal suspects as well as arresting these criminals by investigating inclinations in location, crime type, habit, and additional patterns of behaviors.

Researching field : Data mining can aids researchers by speeding up their process of analyzing the data; therefore, permitting those more time to work on other projects.

II. FREQUENT ITEMSET MINING (FIM)

Frequent Itemset mining is studied for the market basket analysis. It is a kind of data analysis technique that finds the certainties and uncertainties in the data systems. It is mainly predictable for predicting the purchasing behavior. It deals with the products recognition and mail delivery subsystem. In certain cases, multitude of the data assigning tasks to be followed.

Already from the start, Frequent Itemset Mining (FIM) has been an essential part of data analysis and data mining. FIM tries to extract information from databases based on frequently occurring events, i.e., an event, or a set of events, is interesting if it occurs frequently in the data, according to a user given minimum frequency threshold. Many techniques have been invented to mine databases for frequent events. These techniques work well in practice on typical datasets, but they are not suitable for truly Big Data. Applying frequent itemset mining to large databases is problematic. First of all, very large databases do not fit into main memory. In such cases, one solution is to use level-wise breadth first search based algorithms, such as the well known Apriori algorithm, where frequency counting is achieved by reading the dataset over and over again for each size of candidate itemsets. Unfortunately, the memory requirements for handling the complete set of candidate itemsets blows up fast and renders Apriori based schemes very inefficient to use on single machines. Secondly, current approaches tend to keep the output and runtime under control by increasing the minimum frequency threshold, automatically reducing the number of candidate and frequent itemsets. However, studies in recommendation systems have shown that itemsets with lower frequencies are more interesting.

Item Set Enumeration derives the general top-down search scheme for item set enumeration from the fundamental properties of the support measure, resulting in breadth-first and depth-first search, with the sub-problem and item order providing further distinctions. Database Representations reviews different data structures by which the initial as well as conditional transaction databases can be represented and how these are processed in the search. Advanced Techniques collects several advanced techniques that have been developed to make the search maximally efficient, including perfect extension pruning, conditional item reordering, the k-items machine, and special output schemes. Intersecting Transactions briefly surveys intersecting transactions as an alternative to item set enumeration for finding closed (and maximal) item sets, which can be preferable in the presence of (very) many items. Extensions discusses selected extensions of the basic approaches, such as association rule induction, alternatives to item set support, association rule and item set ranking and

filtering methods, and fault-tolerant item sets.

2.1 Working of Frequent Itemset Mining

First, we build the header table which consists of item name and link field corresponding to each item. All link entry of header table is initially set to null. Whenever an item first time added into the tree, the corresponding entry of header table is updated. The root, labeled as "null", is created. Children are added by scanning the database.

TID	*ITEMS*	Frequent Items
T1	A, B	A, B
T2	C, D, E	D, E, C
T3	A, B, C, F	A, B, C, F
T4	A, C, D	A, D, C
T5	A, D, F	A, D, F
T6	B, E, I	B, E
T7	A, B, D, E	A, D , B, E
T8	D, E, F, I	D, E, F

Figure 1. Transaction details

First, a path that shares same prefix is require to be search. If there exist a path that is same as any prefix of current items of transaction (in ist order) then the count of prefix portion is incremented by one in the tree, remaining items of same transaction (which do not share the path), are added from the last node of sharing portion in ist order and their count value is set to 1. If items of a transaction do not share any path of tree then they are added from the root in ist order. Each path of prefix tree (FP-tree) represents a set of transactions.

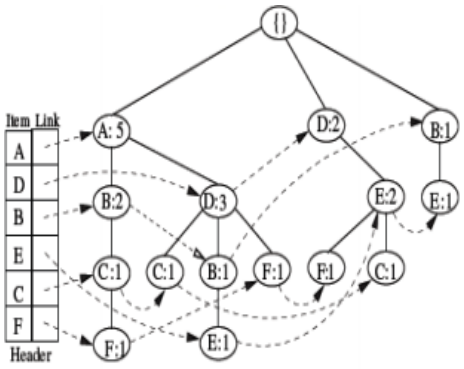

Figure 2. FP tree

The limitations of these existing methods are the ones inherited from the original methods. The size of the data for the level-wise generate-and-test techniques affects their scalability and the pattern-growth techniques require a lot of memory for accommodating the dataset in the data structures, such as the FP-tree, especially when the transactions do not share many items. In the case of uncertain data, not only the items have to be shared for a better compression but also the existence probabilities, which is often not the case.

III. MOTIVATION

The mining tasks are performed on ordered/ unordered database. It is very sensation and significant research system in the data mining process. Several kinds of applications are available for predicting the futuristic scope with the help of past knowledge. The applications such as telecommunication, user activities, crime detection, illness recovery etc are determined from the frequent based analysis. The objective of the itemset mining is to discover the association among those frequently used items. Relied upon the threshold, the high frequencies are sorted and the lists are updated. In some cases, the threshold level is less in number.

In real-life scenarios, market based analysis is used widely studied. Based on the customer's history, the similar profiles are collected and stored. In order to discover the purchasing behavior, these products similarity are studied. Market basket analyses gives retailer proper information regarding related sales on collection of goods basis Customers who buy s bread frequently moreover buy several products associated to bread like milk, butter or jam. It makes intellect that these groups are situated alongside in a retail center in order that customers can contact them rapidly. This sort of connected customer behavior analysis helps to aware about the customers via logical systems.

IV. PROBLEM DEFINITION

In general context, pattern mining is the field of study that discovers the relationships between each item. The frequency may be captured on the basis of similar itemsets, subsequences and the substructures

from homogeneous and heterogeneous data. The frequency should not lead the defined thresholds. The apriori algorithm is purely depends on the association rule mining systems. Based on the candidates and its substructures, the apriori algorithm is defined. It is further divided into two patterns, namely, apriori frequent pattern growth and the Equivalence CLASS transformation (ECLAT). Most of the transactional databases offer variant knowledge discovery process. Though the transposing of the databases is an easier model, the procedure to be carried out is not easy. In this study, the extraction of similar patterns makes use of transposed database method. A tremendous amount of data has been generated by the transposed database model. It composes of large set of resources and objects. Each object contains a set of attributes. In order to minimize the search space, the data are stored onto vertical format.

V. THE HUN-MAX ALGORITHM

Input: DB: a transaction database, min_util: a user-specified threshold

Output: the set of high-utility itemsets

Step-1: First Scan DB

Step-2: Then calculate the TWU for single items;

Step-3: Identify the set ITWU which contains each item i

Step-4:Each item i such that TWU(i) < min_util;

Step-5: Let the total order of TWU ascending values on ITWU be α;

Step-6: Scan DB to build the utility-list of each item i ∈ ITWU

Step-7: Build the Utility Matrix structure;

Step-8: Find Extensions such as (NULL, ITWU, min_util, Utility Matrix);

VI. EXPERIMENTAL RESULTS

The shopping analysis is taken from the public repository as an application scenario to implement. The comparison of experimental analysis is shown on synthetic and real datasets. It describes the result and performance analysis of the proposed algorithm. Java is the programming tool used for validating the proposed algorithm. The Java programming language is a high-level language that can be characterized by all of the following buzzwords:

- ✓ Simple
- ✓ Architecture neutral
- ✓ Object oriented
- ✓ Portable
- ✓ Distributed
- ✓ High performance
- ✓ Interpreted
- ✓ Multithreaded
- ✓ Robust
- ✓ Dynamic
- ✓ Secure

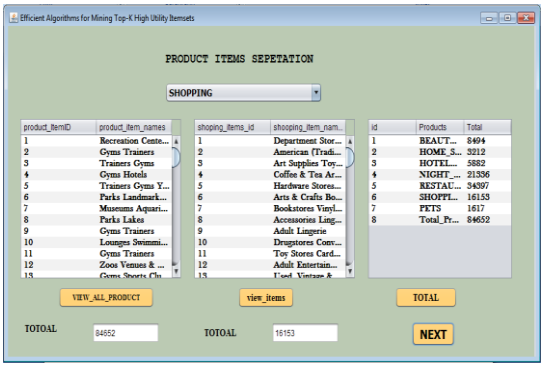

Figure 3. Product Items Separation

Figure 3 represents a loading of the items on real and synthetic datasets in the terms of product items separation. The selected dataset is shopping. There are three tables which contain the overall product, shopping items and category of the product respectively. The first table consists of the entities: product item id and product item name, by which all the products can be viewed like the items are recreation centre, Gym trainers, Gym hotels etc. and the second table is containing two entities: shopping items id and shopping item names by which items can be viewed like Department stores, art supplies toy, hardware stores etc. and the third table contains the entities: id, products, Total. This consequences the total products.

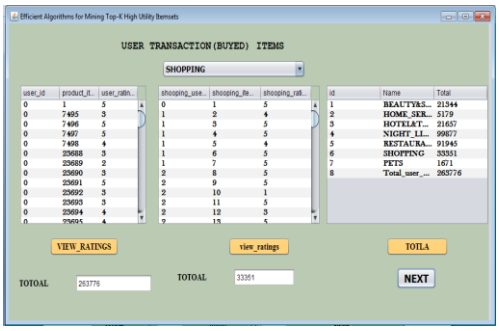

Figure 4. User Transaction (Buyed) Items

Figure 4 represents a loading of the dataset and the selected dataset is shopping. There are three tables; one is containing the user id, Product id and the user rating by which it is used to view the ratings given by the user, second table is containing the entities shopping users, shopping items, shopping ratings, and the last table contains the entities id, names like Beauty & S…, Home_ser, shopping etc. the outcome of this data set is total buyed items.

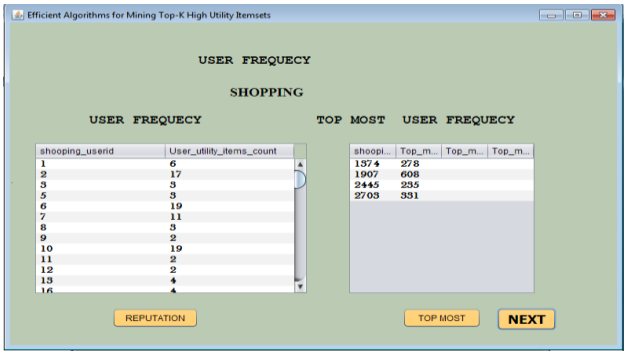

Figure 5. Top most User frequency

Figure 5 represents a loading of the dataset and the selected dataset is shopping. There are two tables; one is for user frequency by including the entities shopping user id and user utility items count by which it consequences the Reputation of the user frequency. Another table is for top most user frequency includes entities shopping items top most user frequency. The outcome of this figure is to show the frequency of top most users.

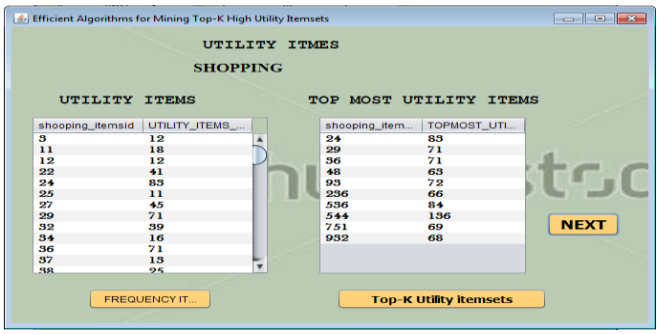

Figure 6. Find out the top k Utility Itemsets

Figure 6 represents a loading of the dataset of the Utility Items. There are two tables; one is for utility items by including the entities shopping items id and utility items and as well as it used to check the frequent items in a dataset. Another table is for top most utility items including the entities shopping items and top most utility. It determines the top k utility items from all the utility items.

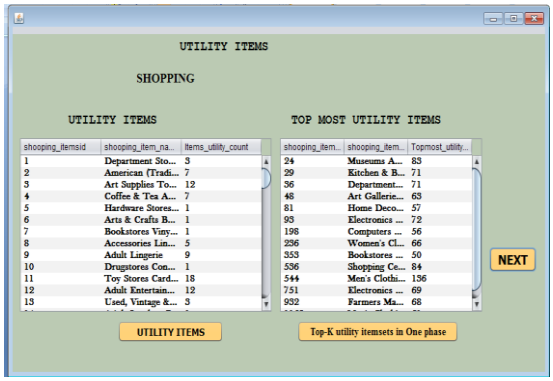

Figure 7. Find out the top k utility itemsets in one phase.

Figure 8. Shopping

Figure 7 represents a loading of the dataset of the Utility Items. There are two tables; one is for utility

items by including the entities shopping items id and shopping items name and items utility count. Another table is for top most utility items including the entities shopping items, shopping items name and top most utility. It determines the top k utility items in one phase from all the utility items.

Figure 8 represents a loading of the dataset of the Utility Items. There are two tables; one is for negative utility items by including the entities shopping items id and shopping items name and topmost utility, the shopping items, in this table are art supplies, drug stores, toy stores, adult entertainment. Another table is for positive utility items including the entities shopping items, shopping items name and items utility. It finds out both the negative and positive utility items.

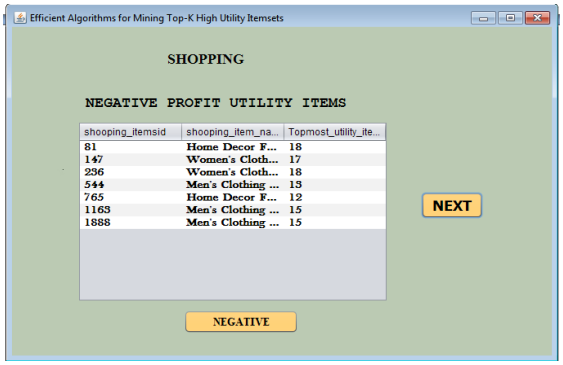

Figure 9. Negative profit Utility Itemset

Figure 9 represents a loading of the dataset of the Utility Items. There is a negative utility profit items dataset. This dataset contains the shopping items id, shopping items name and topmost utility items. The outcome of this is to find the negative profit utility items.

Figure 10. Dataset of the Utility Items

Figure 10 represents a loading of the dataset of the Utility Items. There is a positive utility profit items dataset. This dataset contains the shopping items id, shopping items name and topmost utility items the shopping items are such as museum art gallery, department stores, shopping centers etc. The outcome of this is to find the positive profit utility items.

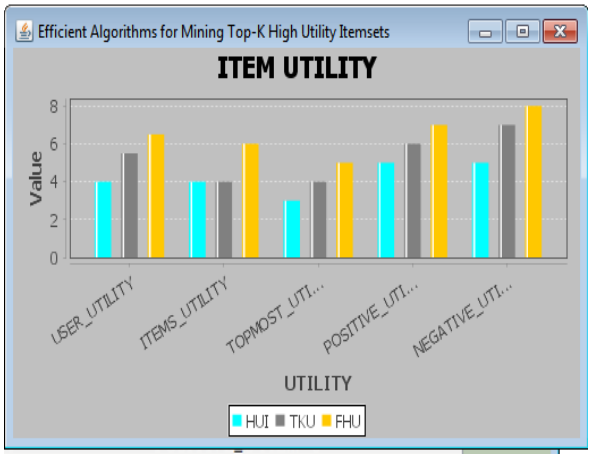

Figure 11. Performance analysis of utility items.

Figure 11 represents the performance analysis in the terms of user utility, items utility, topmost utility, positive utility and negative utility between the proposed HUI (High Utility Items), and existing TKU (Top K Utility) and FHU (Frequently High Utility). It is inferred that the proposed HUI consumes time range of 0-4s than the existing techniques.

VII. CONCLUSION

Frequent itemset mining is one of the recent research study focused by the researchers. In order to limit the size of the output, the itemsets with efficient utilities has to be selected from the pool of resources. Thus, mining of top k itemsets is a tedious task in which k is the required number of itemsets. Based on the characteristics of the users, the threshold value is defined. This study concentrates on developing an efficient top k itemsets with defined min_util thresholds. Two algorithms, namely, mining Top k utility itemsets (TKU) and mining top k utility itemsets in one phase (TKO) is developed for defining min_util threshold value. The mining

performance is enhanced significantly since both the search space and the number of candidates are effectively reduced by the proposed strategies. In the experiments, different types of real datasets are used to evaluate the performance of our algorithm. The experimental results show that TKU outperforms the baseline algorithms substantially and the performance of TKU is close to the optimal case of the state-of-the-art utility mining algorithm.

As a future work, prior techniques have some challenging issues such as, large itemset database required more and more scan iterations which is time consuming task and degrades the efficiency and system performance. Scalability is the major issue as large number of itemsets has been generated during processing. Thus, an efficient solution is required for overcoming dynamic data challenges.

VIII. REFERENCES

[1]. Vincent S. Tseng, Cheng-Wei Wu, Philippe Fournier-Viger, and Philip S. Yu, "Efficient Algorithms for Mining the Concise and Lossless Representation of High Utility Itemsets", IEEE Transactins on Knowledge and Data Engineering, Vol. 27, No. 3, 2015.

[2]. Chowdhury Farhan Ahmed, Syed Khairuzzaman Tanbeer, ByeongSoo Jeong, and Young-Koo Lee, "Efficient Tree Structures for High Utility Pattern Mining in Incremental Databases ", IEEE Transactions on Knowledge and Data Engineering, Vol.21, No 12, December 2009, pp 1708-1721.

[3]. Vincent S. Tseng, Bai-En Shie, Cheng-Wei Wu, and Philip S. Yu, "Efficient Algorithms for Mining High Utility Itemsets from Transactional Databases", IEEE Transactions on Knowledge and Data Engineering, Vol.25, No. 8, AUGUST 2013, pp 1772-1786.

[4]. Chun-Jung Chu, Vincent S. Tseng, Tyne Liang, "An efficient algorithm for mining high utility itemsets with negative item values in large databases", Elsevier, 2009. doi:10.1016/j.amc.2009.05.066.

[5]. Hua-Fu Li, Hsin-Yun Huang, Suh-Yin Lee, "Fast and memory efficient mining of high-utility itemsets from data streams: with and without negative item profits", Springer, 2010. DOI 10.1007.

[6]. Sen Su, Shengzhi Xu, Xiang Cheng, Zhengyi Li, and Fangchun Ya, "Differentially Private Frequent Itemset Mining via Transaction Splitting", IEEE Transactions on Knowledge and Data Engineering, Vol.27, No 7, July 2015

[7]. Vincent S. Tseng, Cheng-Wei Wu, Viger, Philip S. Yu," Efficient Algorithms for Mining Top-K High Utility Itemsets", IEEE Transactions on Knowledge and Data Engineering, DOI 10.1109/TKDE.2015.

[8]. Alva Erwin, Raj P. Gopalan, and N. R. Achuthan, "Efficient Mining of High Utility Itemsets from Large Datasets", In Proc. of PAKDD 2008.

[9]. Shankar, S.; Purusothaman, T.; Jayanthi, S. "Novel algorithm for mining high utility itemsets" International Conference on Computing, Communication and Networking, Dec. 2008.

[10]. Raymond Chan; Qiang Yang; Yi-Dong Shen, "Mining high utility itemsets" In Proc. of Third IEEE Int'l Conf. on Data Mining ,November 2003.

[11]. Ramaraju, C., Savarimuthu N. "A conditional tree based novel algorithm for high utility itemset mining", International Conference on Data mining, June 2011.

[12]. Ying Liu, Wei-keng Liao, Alok Choudhary "A Fast High Utility Itemsets Mining Algorithm" In Proc. of the Utility-Based Data Mining Workshop, 2005.

[13]. Adinarayanareddy B ,O Srinivasa Rao, MHM Krishna Prasad, "An Improved UP-GrowthHigh Utility Itemset Mining" International Journal of Computer Applications (0975-8887) Volume 58-No.2, November 2012.

[14]. 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.

[15]. L. Sweeney, "k-anonymity: A model for protecting privacy," Int. J.Uncertainity Fuzziness Knows.-Base Syst., vol. 10, no. 5, pp. 557–570,2002.

[16]. Y. Bastide, R. Taouil, N. Pasquier, G. Stumme, and L. Lakhal. Mining frequent patterns with counting inference. SIGKDD Explorations Newsletter, 2(2):66 75, December 2000.

[17]. J. Pei, J. Han, and R. Mao. Closet: An efficient algorithm for mining frequent closed itemsets. In DMKD 00: ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, pages 21 30, May 2000.

[18]. M. J. Zaki and C.-J. Hsiao. Charm: An efficient algorithm for closed itemset mining. In SDM 02: Proceedings of the second SIAM International Conference on Data Mining, April 2002.

[19]. K. Gouda and M. J. Zaki. Genmax: An efficient algorithm for mining maximal frequent itemsets. Data Mining and Knowledge Discovovery, 11(3):223 242, 2005.

[20]. G. Grahne and J. Zhu. Efficiently using prefix-trees in mining frequent itemsets. In FIMI 03: Proceedings of the ICDM 2003.Workshop on Frequent Itemset Mining Implementations, November 2003.

[21]. R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," in Proc. Int. Conf. Very Large Data Bases, 1994, pp. 487–499.

[22]. 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.

[23]. K. Chuang, J. Huang, and M. Chen, "Mining top-k frequent patterns in the presence of the memory constraint," VLDB J., vol. 17, pp. 1321–1344, 2008.

[24]. R. Chan, Q. Yang, and Y. Shen, "Mining high-utility itemsets," in Proc. IEEE Int. Conf. Data Mining, 2003, pp. 19–26.

[25]. P. Fournier-Viger and V. S. Tseng, "Mining top-k sequential rules," in Proc. Int. Conf. Adv. Data Mining Appl., 2011, pp. 180–194.