International Journal of Technology Research and Management
ISSN (Online): 2348-9006
Vol 6 Issue 3 March 2019

An Efficient Hash Based Technique for Mining All High Utility Item Sets From a Transaction and Profit Data Set

Ashok Patidar[1], Nitika Kadam[2]
Department of computer science, LNCT, Indore, M.P., India[1, 2]
patidarashok13@gmail.com[1], kadamnitika2@gmail.com[2]

Abstract: *Data mining is the heart favorite topic of research across the globe. New methods and models a have been developed to solve the data mining problems. Utility mining is an important data mining task. It is an extension of Market Basket Analysis. This paper presents a review of work done in utility mining. There are several algorithm have been developed to solve the problem of high utility item set mining. But there is still a lot of scope of work in terms of efficiency. This paper also proposes a hash based technique for utility mining. The proposed technique makes use of hash data structure for storing the data base. This type of storage provides speedy access to data. Experimental study has shown that the proposed work is having less time complexity.*

Keywords: *Data Mining, KDD, High Utility Mining, Minimum Utility.*

1. INTRODUCTION

Data mining is a technique that helps to extract important data from a large database [1]. It is the process of sorting through large amounts of data and picking out relevant information through the use of certain sophisticated algorithms. As more data is gathered, with the amount of data doubling every three years, data mining is becoming an increasingly important tool to transform this data into information.

Data mining extracts novel and useful knowledge from data and has become an effective analysis and decision means in corporation. In utility mining [1] we concentrate on utility value of itemset while in frequent item set mining we concentrate that how frequently items appears in transactional database.

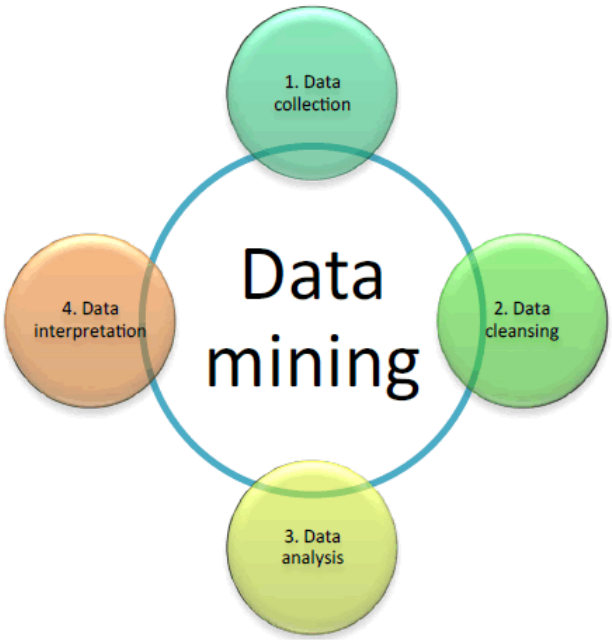

Figure 1: Data-mining process [1,2].

Basic Concepts:

The basic definitions are as follows:

Definition 1: A frequent itemset is a set of items that appears at least in a pre-specified number of transactions.

Formally, let I = {i1, i2, . . . , im} be a set of items and DB = {T1, T2, ..., Tn} a set of transactions where every transaction is also a set of items (i.e. itemset).

Definition 2. The utility of an item ip is a numerical value yp defined by the user. It is transaction independent and reflects importance (usually profit) of the item. External utilities are stored in an utility table.

Definition 3: The utility of an item set X in a transaction Ti is denoted by U(X,Ti) & it is calculated as follows. For example,U({*AC*}, *T1*) = U({*A*}, *T1*) + U({*C*}, *T1*) = 5 + 1 = 6.

Definition 4: The utility of an item set X in D is denoted by U(X) & it is calculated as follows For example, U({*AD*}) = U({*AD*}, *T1*) + U({*AD*}, *T3*) = 7 + 17 = 24.

Definition 5: An itemset is called a *high utility itemset* if its utility is no less than a user-specified *minimum utility threshold* which is denoted as *min_util*. Otherwise, it is called a *low utility itemset.*

Table 1:Transaction Data Set

TID	TRANSACTION	TU
T1	(A,1) (C,1) (D,1)	8
T2	(A,2) (C,6) (E,2) (G,5)	27
T3	(A,1) (B,2) (C,1) (D,6) (E,1) (F,5)	30
T4	(B,4) (C,3) (D,3) (E,1)	20
T5	(B,2) (C,2) (E,1) (G,2)	11

Table 2: Item & correspondent profit

ITEM	A	B	C	D	E	F	G
PROFIT	5	2	1	2	3	1	1

Definition 6: The transaction utility of a transaction Td is denoted as TU(Td) and defined as u(Td, Td). For example, TU(T1) = u({ACD}, T1) = 8.

2. LITERATURE REVIEW

Y. G. Sucahyo at all [2], **constitutes that** CT-PRO is also the variation of classic FP-tree algorithm. It is based upon the compact tree structure. This algorithm uses bottom up approach for performing tree traversal. This is not a recursive method.

ZHOU Jun et al.[3], proposed this algorithm by considering the space as an important factor. Authors used an improved LRU (Least Recently Used) based algorithm. Proposed algorithm omits the infrequent items before taken for the processing. Method increases the stability and the performance. Method is used to find out the frequent items as well as the frequency of those items.

V. S. Tseng et al [4], proposed a FP tree based algorithm, this algorithm uses a tree to maintain the TWU information. It also uses the concept of pruning to eliminate the useless items from the first phase of the algorithm. This pruning helps in saving storage space as the size of tree reduces

Ahmed et al., [1] proposed a tree-based incremental high utility pattern mining (IHUPM) algorithm. In this work a tree based structure called IHUP-Tree which is used to maintain the information about itemsets and their utilities. This work proposes three tree structures to perform incremental and interactive high utility pattern mining efficiently. This reduces the calculations when a minimum threshold is changed or a database is updated.

Junqiang Liu et al., [5] proposed an algorithm direct discovery high utility pattern (D2HUP) which gains the combination of high utility pattern miner and utility pattern. This algorithm mines utility itemset in share framework. The direct discovery of high utility patterns, which is an integration of the depth-first search of the reverse set enumeration tree. This algorithm addresses the scalability and efficiency issues occurred in the existing systems as it directly extracts the high utility patterns from large transactional databases. This algorithm is based on the powerful pruning approaches.

Shankar et al., [6] proposed a fast utility mining (FUM) algorithm that finds all high utility itemset within the given utility constraint threshold. It is faster and simpler than the original UMining algorithm. This algorithm efficiently handles the duplicate itemsets. It checks whether a transaction defined by an itemset purchased in it, repeats its occurrence in a later transaction. If a later transaction also contains same itemset purchased in any of the previous transactions, then that transaction is ignored from processing and duplicate itemset are removed. This reduces the execution time of the algorithm further more.

Lee et al., [7] proposed a high utility itemset miner (HUI-Miner) for high utility itemset mining. This algorithm uses a novel structure called utility-list which is used to store both the utility information about an itemset and the heuristic information for pruning the search space. This algorithm first creates an initial utility list for itemsets of the length 1 for promising items. This algorithm constructs recursively a utility list for each itemset of the length k using a pair of utility lists for itemset of the length k-1 for mining high utility itemset, each utility list for an itemset keeps the information of indicates transaction for all of transactions containing the itemset, utility values of the item set in the transactions, and the sum of utilities of the remaining items that can be included to super itemset of the itemset in the transactions.

Vincent S. Tseng, Fournier-Viger and Philip S. Yu[8] have proposed a new framework for top-k high utility itemset mining, where k is the desired number of HUIs to be mined. Two types of efficient algorithms named TKU (mining Top-K Utility itemsets) and TKO (mining Top-K utility item sets in one phase) are proposed for mining such itemsets without the need to set min_util.

Cheng Zhou, Boris Cule and Bart Goethals [9]addressed the problem of sequence classification using rules composed of interesting patterns found in a dataset of labeled sequences and accompanying class labels. We [10] measure the interestingness of a pattern in a given class of sequences by combining the cohesion and the support of the pattern.

3. PROPOSED ALGORITHM

Step 1: Input:

- A Transaction data Base T & correspondent Profit table P
- Minimum utility value is 6

Table 1: Transaction Data Set (T)

TID	TRANSACTION
T1	A C D
T2	A C E G
T3	A B C D E F
T4	B C E
T5	B C E G

Table 2: Item & correspondent profit (P)

ITEM	A	B	C	D	E	F	G
PROFIT	5	2	1	2	3	1	1

Step 2: We convert the above two table in hash map structure as follows:

TID	Items	Utility Value
T1	A	5
	C	1
	D	2
T2	A	5
	C	1
	E	3
	G	1
T3	A	5
	B	2
	C	1
	D	2
	E	3
	F	1
T4	B	2
	C	1
	E	3
T5	B	2
	C	1
	E	3
	G	1

We scan above table calculate the weighted transaction utility (WTU) of each item WTU(A)= A is pre sent in transaction number T1, T2, T3 in hash map. Also profit of A is 5 as mentioned. So the weighted transaction utility of A is calculated as follows:

WTU(A)= 5+5+5 = 15
WTU(B)= 2+2+2 = 6
WTU(C)= 5
WTU(D)= 4
WTU(E)= 12
WTU(F)= 1
WTU(G)= 2

Now we compare the wtu of each item with minimum utility which is 6 & include only those items in high utility list whose wtu is greater than or equal to the minimum utility.

Now we see that the wtu of A , B, & E is greater then are equal to 6. So A, B & E are included in high utility item list.

Step 3: In this step, we eliminate all those items from the hash map, whose utility is less than the minimum utility. Along with elimination, we also sort items in decreasing order of their utility.

In previous step, we see that item C, D, F & G are not high utility item sets so we eliminate these items from Table 1. Then we get a new table as follows:

TID	Items	Utility Value
T1	A	5
T2	A	5
	E	3
T3	A	5
	E	3
	B	2

T4	E B	3 2
T5	E B	3 2

Step 4: Now the high utility items of size 1 are A, B & E. we use these items to generate candidates items of size 2.

The candidates of size 2 are obtained by finding all possible combinations of A, B & E. these are

AB, BE, AE.

Now we calculate WTU of AB BE and AE by using the updated table 1.

WTU(AB)= AB together are present in transaction number T3 of updated table 1. So wtu of AB is (5 + 2 = 7).

WTU(BE)= present in 3 transactions of updated table 1 (15).

WTU(AE)= 16

Now we compare wtu of all these with minimum utility (6). We see that all these three items are also high utility items so we add these three items in the list of high utility items.

Step 5: Now the high utility items of size 2 are AB, BE & AE. We use these items to generate candidates items of size 3.

The candidates of size 3 are obtained by finding all possible combinations of AB, BE & AE. Only possible combination of size 3 is ABE

Now we calculate WTU of ABE by using the updated table 1.

WTU(ABE)= ABE together are present in only 1 transaction T3 in updated table 1= 5+2+3=10. ABE is also a high utility item because its wtu is greater than minimum utility.

Now we donot have items, which can be combined to generate a larger item so our algorithm terminates here. The complete list of high utility item is as follows:

A
B
A
AB
AE
BE
ABE

4. RESULT ANALYSIS

The existing method is based on the concept of generate and test method. It means that the algorithm first generates all the candidates of size 1 and then performs the pruning according to the minimum utility. Then it generates all the candidates of size 2 and then perform the pruning according to the min utility. The same process is repeated for the subsequent size elements.

The proposed method generates all the candidates of size 1 and then performs the pruning according to the utility. After that it eliminates all the infrequent items of size 1 from the data set to generate a new compact data set. Then this compact data structure is used to generate the subsequent size elements. So it will save time n space.

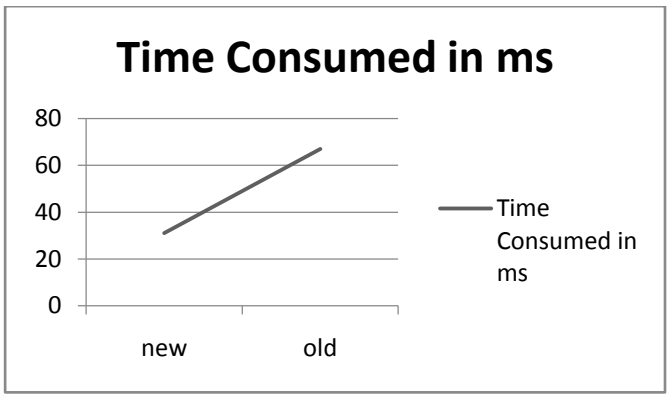

Figure 2: Depicts the Time Consumption Comparison

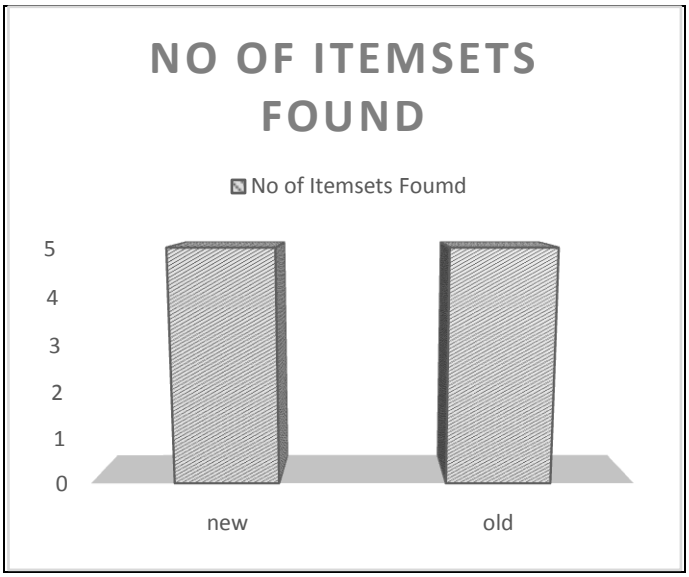

Figure 3: Depicts the Result Comparison

As shown in fig.2 and fig.3 Comparison based on the existing and proposed algorithm. This experiment use a Traffic Accidents Data Set.

5. CONCLUSION

The data capturing technologies is also increasing. In utility mining we concentrate on utility value of itemset while in frequent item set mining we concentrate that how frequently items appears in transactional database. In this paper, we surveyed the list of existing high utility mining techniques. However we surveyed different concepts of Association rule mining and frequent itemset mining techniques which play significant role for basic of utility itemset mining but we restricted ourselves to the classic high utility mining problem. This paper has proposed a time efficient algorithm for mining high utility item sets from a transaction data set.

REFERENCES

[1] C. F. Ahmed, S. K. Tanbeer, B.-S. Jeong, and Y.-K. Lee. Efficient tree structures for high utility pattern mining in incremental databases. In IEEE Transactions on Knowledge and Data Engineering, Vol. 21, Issue 12, pp. 1708-1721, 2009.

[2] Y. G. Sucahyo and R. P. Gopalan. "CT-PRO: A Bottom Up Non Recursive Frequent Itemset Mining Algorithm Using Compressed FP-Tre Data Structure‖. In proc Paper presented at the IEEE ICDM Workshop on Frequent Itemset Mining Implementation (FIMI), Brighton UK, 2004.

[3] ZHOU Jun, CHEN Ming, XIONG Huan A More Accurate Space Saving Algorithm for Finding the Frequent Items.IEEE-2010.

[4] V. S. Tseng, B.-E. Shie, C.-W. Wu, and P. S. Yu, "Efficient algorithms for mining high utility itemsets from transactional databases," IEEE Trans. Knowl. Data Eng., vol. 25, no. 8, pp. 1772–1786, Aug. 2013

[5] Junqiang Liu., Ke Wang., Benjamin., Fung C.M.,"Mining High Utility Patterns in One Phase without Generating Candidates", IEEE Transactions on Knowledge and Data Engineering, Vol. 28, No. 5, pp.1–14, 2016.

[6] Shankar S., Purusothoman T.P, Jayanthi S., Babu N., "A fast agorithm for mining high utility itemsets" ,In Proceedings of IEEE International Advance Computing Conference (IACC), Patiala, India, pp.1459-1464, 2009.

[7] Lee "Top-k High Utility Itemset Mining Based on Utility-List Structures," IEEE Transactions on Knowledge and Data Engineering, Vol. 25, No. 8, pp. 1772–1786, 2016

[8] Vincent S. Tseng, Senior Member, IEEE, Cheng-Wei Wu, Philippe Fournier-Viger and Philip S. Yu, Fellow, IEEE, "Efficient Algorithms for Mining Top-K High Utility Item sets", IEEE Transactions on Knowledge and data engineering, vol. 28, no.1, January 2016.

[9] Cheng Zhou, Boris Cule, and Bart Goethals "Pattern Based Sequence Classification", IEEE Transactions on knowledge and Data Engineering, vol. 28, no. 5, May 2016.

[10] V Kavitha and BG Geetha "Review on High Utility Mining Algorithms", IEEE Conference on Futuristic Trends in Research and Innovation for Social Welfare (WCFTR'16).