Comparison between High Utility Frequent Item Sets Mining Techniques

Mrs. Hetal M. Shah[1] Prof. Bhavesh A. Oza[2]
[2]Professor
[1,2]Department of Computer Science & Engineering
[1,2]L.D. College of Engineering Ahmedabad, GTU-Gujarat, India

Abstract— Data Mining can be defined as an activity that extracts some new nontrivial information contained in large databases. Traditional data mining techniques have focused largely on detecting the statistical correlations between the items that are more frequent in the transaction databases. Also termed as frequent itemsets mining, these techniques were based on the rationale that itemsets which appear more frequently must be of more importance to the user from the business perspective .In this paper we throw light upon an emerging area called Utility Mining which not only considers the frequency of the itemsets but also considers the utility associated with the itemsets. The term utility refers to the importance or the usefulness of the appearance of the itemset in transactions quantified in terms like profit , sales or any other user preferences. This paper presents a novel efficient algorithm FUFM (Fast Utility-Frequent Mining) which finds all utility-frequent itemsets within the given utility and support constraints threshold. It is faster and simpler than the original 2P-UF algorithm (2 Phase Utility-Frequent), as it is based on efficient methods for frequent itemset mining. Experimental evaluation on artificial datasets shown here, in contrast with 2P-UF, our algorithm can also be applied to mine large databases.

Key words: Frequent Pattern Mining, 2PUF, FUFM, Quasi support, extended support

I. INTRODUCTION

Data mining is a very useful in marketing to analyze the data and to predict the future. Data Mining can be defined as an activity that extracts some new nontrivial information contained in large databases.[1]. Different Data Mining techniques and algorithms are used to extract information from large amount of data. These techniques include Association rule mining, Prediction techniques etc. The association rule mining derives some rules which describe the relationship between item sets. Standard methods for association rule mining are based on support and confidence measures. The goal of the first phase of association rule mining is to find all frequent itemsets and the goal of the second phase is to build rules based of frequent itemsets. We use support measure because we assume that the user is interested only in statistically important patterns. The prediction techniques helps to predict the future based on these association rules. This leads to better decision making about the future.

The researchers were concentrated on the items which are frequently purchased by the customers. Different algorithms were proposed to find the frequent item sets like Apriori [1, 2, 3], FP-TREE etc. Later their interest moved to the high utility item sets. Utility may be in any form like profit, cost, sales or any other user preferences. An item set is said to be a high utility item [3], if the utility of that item is greater than or equal to user specified utility. The utility is obtained by the product of internal utility and external utility. The internal utility is the quantity of each item and the external utility is the profit obtained on that item. The most usual form that is also used in this paper is defined as a sum of products of internal and external utilities of present items. The goal of high utility itemset mining is to find all itemsets that give utility greater or equal to the user specified threshold.

The retail organizations stores large amounts of data regarding their sales and customers in databases. Later they view those databases and extract the information which they are interested in. Some organizations may interest in the frequent itemsets purchased by the customers, some may interested in the items which gives more profit to the organization. The organizations extract the information required to them based on their interest from the databases. This information helps the organization later, to take decisions to improve their market.

II. TECHNIQUES FOR UTILITY PATTERN MINING

Most Organizations are interested in the items which gives more profit and which are purchased frequently by the customer are called as utility frequent item sets. The utility item sets are those which give more profit on the items. The frequent items are those which are frequently purchased by the customers without considering their profits. There are various techniques are proposed for generating utility frequent item sets so that high profitable items are mined efficiently. Here is the comparison between two basic techniques.

- 2PUF Algorithm
- FUFM Algorithm

A. 2PUF Algorithm:

First algorithm 2P-UF for mining utility-frequent itemsets was introduced together with formal definition of this novel area [4] of utility-based itemset mining. It is based on a quasi support, a special measure that solves the problem of nonexistence of anti-monotone property of joined support-utility measure. 2P-UF is proven to find all utility-frequent itemsets but it has some properties that render it impossible to use in practice on large databases.

To find the utility frequent item sets, 2P-UP [8], **Internal utility** [3, 8] refers to the quantity of items and **external utility [5]** refers to the profit of each item. The item sets are high utility item sets if their utility is greater than or equal to a user specified extended support value. The item sets are frequent if the percentage of the support count in a database is greater than or equal to user specified support threshold. The utility can be measured by the product of **internal utility and external utility.**

Example:

TID	A	B	C	D	E
T1	1	0	10	1	0
T2	2	0	6	0	2
T3	2	2	0	6	2

T4	0	4	13	3	1
T5	0	2	4	0	1
T6	1	1	1	1	0

Table1 Transaction Table

Utility of Database and Transactions:
U[DB]= U[T1]+U[T2]+U[T3]+U[T4]+U[T5]+U[T6]
U[T1]= 1*5+0+10+1*2=5+10+2=17
U[T2]= 10+0+6+0+6+5+0=27
U[T3]= 10+4+0+12+6+5+0=47
U[T4]= 8+13+6+3=30
U[T5]= 4+4+9+2=19
U[T6]= 5+2+1+2+2=12
U[DB]= 17+27+47+30+11+12=144
Support calculation: Total items=7
Take 4 itemsets
{ABCD}
{ABCE}
{BCDE}
{ABDE}
{ACDE}
Support threshold=0.63
Quasi:
{ABCD}
T1= 1*5+1*10+1*2= 5+10+2 = 17
T2= 2*5+6*1=10+6= 16
T3= 2*5+2*2+6*2=10+4+12= 26
T4= 4*2+13*1+3*2=8+13+6= 27
T5= 2*2+4*1=4+4= 8
T6=1*5+1*2+1*1+1*2= 5+2+1+2= 10
Quasi support {ABCD} =17+16+26+27+10+8/144
= 0.72
FOR ITEMSET {ABCE}
T1= 1*5+10*2= 25
T2= 2*5+6*1+2*3=22
T3= 2*5+2*2+2*3=20
T4= 4*2+13*1+1*3=24
T5= 2*2+4*1+1*3=11
{ABCE}= 22+20+24+11+8/144 =0.590
FOR ITEMSET {BCDE}
T1= 10*1+2*1= 12
T2= 6*1+2*3= 12
T3= 2*2+6*2+2*3= 22
T4= 4*2+13*1+2*3+1*3= 30
T5= 2*2+4*1+1*3=11
{BCDE}=12+12+22+30+11+5/144 =0.63
FOR ITEMSET {ABDE}
T1= 1*5+1*2= 7
T2= 2*5+2*3= 16
T3= 2*5+2*2+6*2+2*3= 32
T4= 4*2+3*2+1*3= 17
T5=2*2+1*3= 7
{ABDE}= (7+16+32+17+7+9)/144 =0.611
FOR ITEMSET {ACDE}
T1= 1*5+10*1+1*2=17
T2= 2*5+6*1+2*3=22
T3= 2*5+6*2+2*3= 28
T4= 13*1+3*2+1*3= 21
T5= 4*1+1*3= 7
{ACDE}= (17+22+28+21)/144 =0.680
PERFORM INTERSECTION OPERATION ON SETS
{ABCD}{BCDE}{ACDE}
{ABCD} ^ {ACDE}={ACD}
{ABCD} ^ {BCDE}={BCD}
{BCDE } ^ {ACDE}={CDE}
FOR ITEMSET {ACD}
T1=1*5+10*2+1*2=27
T2=2*5+6*1=16
T3=2*5+6*2=22
T4=13*1+3*2=1
T5=4*1=4
T6=1*5+1*1+1*2=8
{ACD}= (27+16+22+1+4+8)/144 =0.66
FOR ITEMSET {BCD}
T1=10*1+1*2=12
T2=6*1=6
T3=2*2+*2=16
T4=4*2+13*1+3*2=27
T5=2*2+1*1+1*2=5
T6=2*1+1*1=1*2=5
{BCD}= (12+6+16+27+8+5)/144 =0.506
FOR ITEMSET {CDE}
T1=10*1+1*2=12
T2=6*1+2*3=12
T3=6*2+2*3=18
T4=13*1+3*2+1*3=21
T5=4*1+1*3=7
T6=1*1+1*2=3
{CDE}= (12+12+18+21+7+3)/144 =0.506
PERFORM INTERSECTION OPERATIONS:
{ACD} ^ {BCD}={CD}
{BCD} ^ {CDE}={CD}
{ACD} ^ {CDE}={CD}
FOR IITEMSET {CD}
T1=10*1+1*2=12
T2=6*1=6
T3=6*2=12
T4=13*1+3*2=19
T5=4*1=4
T6=1*1+1*2=3
{CD}= (12+6+12+19+4+3)/144 =0.361
Therefore the high utility itemsets are:
{ABCD}
{BCDE}
{ACDE}
{ACD}

1) *Disadvantage of 2P-UF algorithm:*
   - The reversed way of candidate generation 2P-UF algorithm wastes time checking long itemsets that are highly unusual to be utility-frequent.
   - Candidate generation function is also slow and inefficient as it computes intersection of every pair of candidates in each iteration.
   - Computation of quasi support measure is also inefficient because special data structures (hash trees) cannot be used and we have to scan database once for every candidate.
   - The two-phase form of the algorithm is space consuming since we have to store all quasi utility-frequent candidates from the first phase to filter them in the second phase.

2) *Advantage of 2P-UF algorithm:*
   - 2P-UF algorithm is efficient only in case when utility threshold is very high and result is an empty

set. In such case the mining process stops at the very first iterations.

B. FUFM Algorithm:

Fast Utility Frequent Mining Algorithm overcomes the disadvantages of the 2P-UP algorithm. The algorithm starts with one item set generation and it finds the set of candidates with support greater than or equal to user specified minsup. Then compute extended support for all candidates. The extended support measure can be calculated as,

Support(I,u) = $\left| T_{I,u} / D \right|$,

where $T_{I,u}$ = {T $\left|\right.$ I C T' ^ u(I,T)≥ μ ^ T Є D }

Example:

Consider the following database:

minsupp=2,utility threshold =1,support threshold=0.1

TID	A	B	C	D	E
1	0	0	18	0	1
2	0	6	0	1	1
3	2	0	1	0	1
4	1	0	0	1	1
5	0	0	4	0	2
6	1	1	0	0	0
7	0	10	0	1	1
8	3	0	25	3	1
9	1	1	0	0	0
10	0	6	2	0	2

Table1: Transaction Table

Item	A	B	C	D	E
Profit on each Item	3	10	1	6	5

Table 2: Profit Table

ALGORITHM: GENERATE 1-ITEM SET:

ITEM	COUNT
A	8
B	24
C	50
D	6
E	10

GENERATE ITEM SETS THAT SATISFIES MINSUP=2

ITEM	COUNT
{A}	8
{B}	24
{C}	50
{D}	6
{E}	10

CALCULATE EXTEND SUPPORTS OF EACH ITEM SET:

ITEM	**COUNT**
{A}	0.057
{B}	0.57
{C}	0.12
{D}	0.08
{E}	0.12

1) Advantage of FUFM algorithm:

- FUFM algorithm does not have disadvantages and inefficiencies of the 2P-UF algorithm as its generation phase is based on frequent itemset mining methods.
- Filtering non-utility frequent candidates is also efficient because we only need to build a hash tree from candidates and push all transactions down the tree to compute subsets.
- Consequently, time and space complexity are both fully determined with the complexity of the frequent itemsets mining method used.

III. Conclusions and Future Work

Association rule mining is the most popular data mining algorithm. Many number of efficient techniques available for association rule mining, which considers mining of frequent itemsets. But an promising technique in Data Mining is Utility Mining, which incorporates utility considerations during itemset mining. Utility Mining covers all aspects of economic utility in data mining. In this paper, fast algorithm for mining all utility-frequent itemset. It is considerably faster than first algorithm 2P-UF and also much simpler to implement. Because it is based on efficient methods for mining frequent itemset it also performs well on real-sized databases. In the future work, new algorithm will be proposed predicting and classifying the customers based by maintaining customer ids for improving the business.

References

[1] Aakansha Sexena, Sohil Gandhiya, " A Survey on Frequent Pattern Mining Methods Apriori, Eclat, Fpgrowth",2014 IJDER|Volume 2, issue|ISSN 232-9939

[2] Varsha Mashoria, Anju Singh,"Literature Survey on Various Frequent Pattern Mining Algorithm", Vol 3, issue1 (Jan 2013), pp58-64.

[3] V.S. Tseng, C.-W. Wu, B.-E. Shie, and P.S. Yu, "UP-Growth: An Efficient Algorithm for High Utility Itemsets Mining," Proc. 16th ACM SIGKDD Conf. Knowledge Discovery and Data Mining (KDD'10), pp. 253-262, 2010.

[4] Yeh J. S., Li, Y. C., Chang C. C.: A Two-Phase Algorithm for Utility-Frequent Mining. To appear in Lecture Notes in Computer Science, International Workshop on High Performance Data Mining and Applications, 2007.

[5] Y. Liu, W. Liao, and A. Choudhary, "A Fast High Utility Itemsets Mining Algorithm," Proc. Utility-Based Data Mining Workshop,2005.

[6] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules," Proc. 20th Int'l Conf. Very Large Data Bases (VLDB), pp. 487-499, 1994.

