Efficient Mining of High Utility Sequential Pattern from Incremental Sequential Dataset

Uma Dave
Computer Engineering Dept.
Sardar Vallabhbhai Patel
Institute of technology,
Vasad, Gujarat, India

Jayna Shah
Assistant Professor
Computer Engineering Dept.
Sardar Vallabhbhai Patel
Institute of technology,
Vasad, Gujarat, India

ABSTRACT

Frequent Pattern mining is modified by Sequential Pattern Mining to consider time regularity which is further enhanced to high utility sequential pattern mining (HUS) by incorporating utility into sequential pattern mining for business value and impact. In the process of mining HUS, when new sequences are added into the existing database the whole procedure of mining HUS starts from the scratch, in spite of mining HUS only from incremental sequences. This results in excess of time as well as efforts. So in this paper an incremental algorithm is proposed to mine HUS from the Incremental Database. Experimental results show that the proposed algorithm executes faster than existing PHUS algorithm resulting in saving of time as well as efforts.

General Terms

Sequential Pattern Mining, Utility Mining, High utility Sequential pattern Mining

Keywords

QSD, su, suub , asu , hus

1. INTRODUCTION

Association rule mining is a popular data mining task which is essential to a wide range of practical applications viz. Supermarket promotions, biomedical applications etc. However, time regularity of items in the databases cannot be found by using traditional association rule mining approaches. So the concept of Sequential Pattern Mining has emerged (Agrawal and shrikant, 1995) for discovery of regularity knowledge which has proven to be very essential for order based critical business problems such as behaviour analysis, gene analysis, web log mining and DNA sequences.

Basically Sequential pattern Mining is the process of finding complete set of frequent subsequence that is the subsequence whose occurrence frequency in the sequence database is greater than or equal to minimum support[14]. So it considers not only of the frequency relationships of items in the pattern but also the order relationship of items according to timestamp of items. For example assume a sequential pattern like "< (Computer), (Printer) > ". The pattern states that most customers will buy a printer in a period of time after they have bought a computer.

However the pattern found by Sequential pattern Mining techniques do not reflect any other factors such as cost, price or profit. Assume another pattern"< (Diamond),(Necklace) > " in a sequence database. Assume it is not high frequency pattern in the database. So it is not identified by the process of Sequential pattern Mining but the pattern may contribute a large portion to the overall of the shop due to its high profit. Accordingly some product with high profit but low frequency may not be discovered by traditional approaches.

So to address the above problem a new concept called "Utility" has been introduced in Sequential Pattern Mining which considers not only quantity of items in a sequence but also the individual profit of items in a quantitative sequence database. A sequence is of high utility, if its utility is greater than or equal to predefined minimum utility threshold [17]. The major goal of high utility sequential pattern mining is to mine high utility sequential purchasing so that the identified shopping sequences are more informative for retailers in determining their marketing strategy.

However in the process of mining high utility sequential pattern, the database was assumed to be static but in real-world applications, sequences in a database do not usually remain in a stable condition. In real field new sequences gets generated and added gradually in existing sequence database due to new purchases from customers. So in order to mine high utility sequential pattern for incremental sequences, the whole procedure of mining has to rescan the existing data followed by scanning incremental sequences. This not only involves the repeated scanning efforts but also results in wastage of time as well as efficiency.

As the procedure for mining high utility sequential pattern can't give solution to the above stated problem, a new approach called mining of high utility sequential pattern from incremental sequential database has been emerged.

2. REVIEW OF RELATED WORK

In this section, some related studies on utility mining and high utility sequential pattern mining are briefly reviewed.

2.1 Utility Mining

Utility can be defined as the how useful or profitable an itemset is. The meaning of itemset utility is interestingness, importance or profitability of an item to its users[9]. In practice the utility value of an itemset can be profit, popularity, page rank, measure of some aesthetic aspect such as beauty or design o some other measures of user's preferences [10]. Basically utility are of 2 types: 1) Internal utility, 2) External utility. Internal utility represents quantity of item in a transaction and external utility represents unit profit of each item. Multiplication of internal and external utility produces the value of utility. The itemset is of high utility if its utility is greater than or equal to predefined minimum utility threshold.

Several algorithms are available for mining high utility itemsets. In 2004, Umining algorithm for mining high utility itemset has been proposed. But the problem with this is that, it cannot extract complete set of them. In frequent sequential pattern mining downward closure property serves as the foundation of pattern mining algorithm. However this property doesn't hold in high utility pattern mining problem. So transaction weighted closure property was introduced, in which a two phase algorithm was proposed[4]. As compared

to Umining, two phase algorithm was faster and more efficient. However when new transactions are inserted into original database, the whole procedure of mining starts from scratch. So an incremental algorithm for mining high utility itemsets was proposed.

Table 1: The Set of 7 Quantitative Sequences

SID	QUANTITATIVE SEQUENCES
Seq_1	<a(1),{a(2),d(1)},e(1),f(2)>
Seq_2	<a(1),{b(1),c(15)} ,c(3)>
Seq_3	<a(3),{a(2),c(8) },e(2)>
Seq_4	<{a(3),b(2)},c(2),e(2),f(3)>
Seq_5	<a(3),d(1),f(1)>
Seq_6	<a(2),c(4),e(2)>
Seq_7	<e(3),c(2),e(1)>

2.2. Problem Statement &Definitions:

To clarify describe the problem to be solved , assume a QSD is given in table 1, in which each quantitative sequence consist of Two features, sequence consist of two features sequence identification (SID) and item purchased. There are 6 items in the quantitative sequences denoted as a to f respectively. The value attached to each item in the corresponding slot is the quantity sold in sequence. Also, assume the profit value table 2.

2.3 High utility Sequential Pattern mining from Incremental Database

The concept of high utility sequential pattern mining was extended from the principle of utility mining (Ahmed et al., 2010). This new concept not only considers quantities and profit of items in sequences but also the order relationship of items[7]. According to Ahmed et al.'s definitions (Ahmed et al., 2010), there existed three kinds of calculation for identifying the utility of a pattern in a sequence under corresponding conditions of the three ways. However, the utilities of patterns in quantitative sequences (Ahmed et al., 2010) are difficult to be calculated since a pattern in a sequence has to firstly determined which conditions the pattern belongs to. To simplify the utility calculation of subsequences in sequences, already developed maximum utility measure[9] is used. For example, assume there exists a quantitative sequence <{(a,2)(b,2)}, {(a,2)(b,3)}, {(a,8)(b,2)}, {(a,3)(b,4)}> , and the profits of the two items, a and b, are 3 and 20. According to Ahmed et al.'s definitions , the utility of <{ab}> is 265. However the utility of <{ab}> by maximum measure is 89. So it is lesser than the Ahmed's utility measure. However, since the main purpose of sequential pattern mining is to find most customers' purchase behavior, but not personal behavior,the maximum utility function may be more appropriately used to find high utility sequential patterns.

However, in the process of mining high utility sequential patterns the database was assumed to be static. However in real field, new data gets generated and added gradually in the existing data in a continuous manner. So, in 2010, the concept of mining high utility itemset from incremental database was introduced but the problem with this is that it mines high utility itemsets from transaction database only. So, an incremental algorithm is proposed which mines high utility sequential patterns from incremental sequential database. Below related definitions of high utility sequential pattern mining mining are described as under.

Table 2 : Predefined Profit Values of the items

ITEM	a	b	c	d	e	f
PROFIT	3	10	1	6	5	2

Definition 1

An itemset X is a subset of items sorted alphabetically, X U I. If | X|=r, the itemset X is called an r-itemset. I ={ i1,i2,... ,im} is a set of items, which may appear in sequences. For example, the itemset {ab} contains 2 items and is called a 2- itemset[8].

Definition 2

Seq = <X1,X2,... ,Xm> is a list of itemsets sorted in ascending order according to their occurrence time. Each itemset appears in a sequence Seq is called an element of the sequence. For example, in Table 1, the first element in the fourth sequence Seq8 includes the two items, a and b, and the third element includes the item e[7].

Definition 3

A quantitative sequence database QSD is composed of a set of sequences. That is, SDB ={Seq1,Seq2,... ,Seqy,... ,Seqz}, where Seqy is the y-th sequence in QSD[7].

Definition 4

The local sequence utility or internal utility represents the quantity of an item in a sequence. For example, in Table 1, there are three elements in the fifth sequence Seq5, and their local sequence utility or internal utility values are 3,1 and 1, respectively[8].

Definition 5

The external utility of an item represents the profit value associated with each item. For example, the external utility of item a is 3 in Table 2[8].

Definition 6

The utility of an item in a sequence is the multiplication of external utility of the item in the utility table and internal utility of the item in the sequence. Note that the maximum value among the utility values of the item in the sequence is regarded as the utility of the item in that sequence by maximum utility measure. For example, in Table 1, the item a appears two times in Seq1, and the utility values of the item a in the two times can be calculated as 3 (=3 * 1) and 6 (=3* 2), respectively. For the two utility values, the maximum utility value is 6, and the value of 6 is regarded as the utility of the item a in Seq1.[8]

Definition7

The utility of a subsequence in a sequence is the maximum value among utility values of all combinations in the sequence. For example, in Table 1, there are two combinations of the subsequence <ac> in the second sequence Seq2, and the utility values of the two combinations are calculated as 18 (= 3 * 1 + 1* 15) and 6 (= 3 *1 + 1*3), respectively. The value of 18 is regarded as the utility of the subsequence <ac> in Seq2 by the maximum measure[7].

Definition8

The actual sequence utility (asu) of a subsequence in a quantitative sequence database QSD is the summation of the maximum utility values of subsequence in all the sequences. For example in Table 1, $asu_{<ac>}$ = $u2,_{<ac>}$ + $u3,_{<ac>}$ + $u4,_{<ac>}$ + $u6,_{<ac>}$ = 18 + 17 +11 +10 = 56[7].

Definition 9

Let λ be a pre-defined minimum utility threshold. A subsequence S is called a high utility sequential pattern (abbreviated as HUS) if $asu_{(S)} \geq \lambda$. For example in Table 1, if λ =55, then the subsequence <ac> is a high utility sequential pattern since asu $_{<ac>}$ ≥λ(55)[8].

3. PROPOSED MINING ALGORITHM

In this section, an efficient high utility sequential pattern mining algorithm to find high utility sequential patterns from incremental sequential database is proposed. Here the improved indexing strategy is used to help the proposed algorithm.

4.1 Indexing Strategy

In this study, an effective index structure is designed for finding high utility sequential pattern from incremental quantitative sequential database. The structure is an index table(IT) with the 4 fields : The index Value, item/pattern, position vector of each item/pattern and actual sequence utility. The position vector of each item is represented as <sequence-ID, Maximum quantity of item in that sequence>. By using position vector, the proposed incremental algorithm doesn’t need to perform multiple data scans to construct the corresponding projected database of each item/pattern. The actual sequence utility of each item/pattern is calculated by just using the position vector fields. Also when new sequences are inserted into the existing database, only the position vector fields of Index table is modified for finding high utility sequential patterns for Incremental sequences. So it improves the time by reducing the number of database scan and by reducing the computation time required for calculation of actual sequence utility.

Consider the QSD consist of 2 sequences as shown in table 3, and table 4 shows the entry of Index table after scanning 2 sequences of table 4.

Table 3 The Set of 2 Quantitative Sequences

SID	QUANTITATIVE SEQUENCES
Seq_1	<a(1),{a(2),d(1)},e(1),f(2)>
Seq_2	<a(1),{b(1),c(15)} ,c(3)>

4.4 The proposed projection based Incremental Algorithm

In this work, the main concept of proposed approach is extended from the PHUS algorithm[8]. By using this approach, the proposed algorithm does not need to rescan the original sequences for finding high utility sequential patterns from incremental sequences. In addition to quickly find relevant sequences in the mining process, an indexing strategy also helps in reducing the computation time required for computation of actual sequence utility. The detail of projection based incremental algorithm is described as below.

Table 4 : Index Table

Index Value	ITEM	POSITION VECTOR	asu
Null	<a>	{1: 2}{ 2:1}	(2+1)*3=9
	<b>	{2:1}	1*10=10
	<c>	{2:15}	15*1=15
	<d>	{1:1}	1*6=6
	<e>	{1: 1}	1*5=5
	<f>	{1: 2}	2*2=4

Input

A set of items, each with a profit value; a quantitative sequence database QSD, in which each sequence includes a subset of items with quantities; a minimum utility thresholdλ.

Output

A final set of high utility sequential patterns, HUS.

STEP 1

QSD=Incremental Database

STEP 2

For each yth quantitative sequence seq_y in QSD, do the following sub steps,
For each item i_{yz} inseq_y

a) Fetch entry for i_{yz} in Index Table for index value null

If found then,

If position vector for that item in seq_y is empty

Then add position vector (seq-id_y, quantity) of that item in IT.

Else check the quantity of that item

If it is higher than update quantity for seq_y

Else

Add element in position vector for current item i’ along with (Sequence Id, Quantity) in Index Table.

b) Calculate the utility value U_{yz} of the item i_{yz} and add this utility value to the sequence utility su_y of the sequence seq_y.

STEP 3

For each item i in index table,

$asu_{(i)} = 0$;

For each element E in position vector

1) asu(i) = asu(i) + utility (quantity in E *profit of item i)

2) Update the asu(i) in Index table for item i

STEP 4

For each item i in IT, if asu(i) ≥ λ then put it in the set of HUS_1

STEP 5

Process each item i by the following substeps :

a) Set r=1, where r represent the current set of candidate sub sequences to be processed.

b) Fetch the sequence ID of item i from position vector and find the projected sequences of item i ($sd_{(i)}$) from QSD.

c) Find all high utility sequential patterns with i as their prefix item by finding HUS(i, $sd_{(i)}$, r) procedure. Let the set of returned high utility sequential pattern as HUS.

STEP 6

Output the final set of high utility sequential pattern in all the HUS_i .

Finding- HUS< i, $sd_{(i)}$, r) procedure :

Input : A prefix r-subsequence, i and its projected sequences $sd_{(i)}$

Output : The high utility sequential patterns with i as their prefix.

PSTEP1 : Do the following sub steps for each for each seq_yin $sd_{(i)}$.

(a) Get each item i located after the prefix r-pattern in seq_y

(b) Generate the r+1 subsequence i' composed of the prefix r-pattern i & put that pattern i' in

(c) the Index table with index value i,

If i' already exist in index table, fetch position vector for seq_y

If position vector for that item in seq_y is empty, than add position vector for that (seq-id_y , quantity)of item in IT.

Else check the quantity of that item, if it is higher than update the quantity for seq_y

Else

Add i' in index table along with thepositionvector,vector element (Sequence ID, quantity).

PSTEP 2: For each subsequence i' in the Index tablewithiasPrefix (Index value) ,

Calculate the asu(i') of the subsequence usingposition vector

If asu(i') of the subsequence i' is greater than or equal to threshold then put in the set of HUS_{r+1} with i as their prefix

PSTEP 3

For each subsequence i' in the Index table with i as their prefix (index value)

Find the projected sequences including i' in $sd_{(i')}$ by using IT where i' and the items located after i in each sequence are projected in the new sequence and put it in $sd_{(i')}$

PSTEP 4

Find all the high utility sequential patterns with i' as their prefix patterns by finding-HUS (i', $sd_{(i')}$, r+1) procedure.Let the set of high utility sequential patterns returned be $HUS_{i'}$.

PSTEP 5

Return the set of HUS in all the HUS_i.

4. EXAMPLE OF MINING HUS FROM INCREMENTAL DATABASE

Here, a simple example is given to show how the proposed algorithm can be used to find high utility sequential pattern from Incremental Sequential Database.

Initially assume there are 7 sequences shown in table 1 for mining and there are 6 items in the sequences respectively denoted as a to f. Also assume that the profit value for each single item is predefined in table 2. Moreover the minimum utility threshold λ is set as 65. The proposed algorithm proceeds as follows to find high utility sequential pattern from the sequence data in table 1.

STEP 1: Sequences are scanned in order of Sequence ID (SID) and entries in the Index table are updated accordingly as shown in table 5. For example, consider the first sequence. The first item in the seq_1 is a, which is initially not present in Index Table(IT) , so item a is entered in IT along with position vector as <1:1>. Now moving ahead, the next item is also a, but item a has a quantity higher than the previous one so the position vector field of a is updated as <1:2> .After that item d is scanned which is not present in IT so new entry in IT is created as d <1:1> .Similarly the whole sequences is scanned and IT is built. Table 5 shows the entries of IT after scanning the 7 sequences.

Table 5: Index Table after scanning 7 sequences of table 1

ITEM	POSITION VECTOR	ASU
<a>	{1: 2} { 2:1} {3:3} {4:3} {5:3} {6:2}	
<b>	{2:1}{4:2}	
<c>	{2:15}{3:8}{4:2}{6:4} {7:2}	
<d>	{1:1}{ 5:1}	
<e>	{ 1: 1} {3:2}{4:2}{6:2}{7:3}	
<f>	{ 1: 2} {4: 3}{5:1}	

STEP 2 : The asu of each item in IT is calculated by using position vector. For example, consider the item <a> whose asu is (2+1+3+3+3+2) *3(Profit value) which is 42. The table 6 shows the result of calculation of asu.

Table 6: Index Table with asu calculation

ITEM	POSITION VECTOR	ASU
<a>	{1: 2} { 2:1} {3:3} {4:3} {5:3} {6:2}	42
<b>	{2:1}{4:2}	30
<c>	{2:15}{3:8}{4:2}{6:4} {7:2}	31
<d>	{1:1}{ 5:1}	12
<e>	{ 1: 1} {3:2}{4:2}{6:2}{7:3}	50
<f>	{ 1: 2} {4: 3}{5:1}	12

STEP 3: Here the HUS_1 is empty because no item in table 6 has asu greater than or equal to λ(65).

STEP 4: After the IT is prepared, the items in IT are sequentially processed. Starting with item a, the sequences in the projected database $sd_{<a>}$ of a is find by examining the position vector of IT. Here the position vector field of IT contains sequences 1,2,3,4,5,6. So these sequences are fetched from the original database. The table 7 shows the result of projected database of <a>.

Table 7: Sequences in the projected database of <a>

SID	QUANTITATIVE SEQUENCES
Seq$_1$	<a(1), { a(2),d(1) },e(1),f(2)>
Seq$_2$	<a(1),{b(1),c(15)} ,c(3)>
Seq$_3$	<a(3),{a(2),c(8) },e(2)>
Seq$_4$	<{a(3),b(2)},c(2),e(2),f(3)>
Seq$_5$	<a(3),d(1),f(1)>
Seq$_6$	<a(2),c(4),e(2)>

All the high utility sequential pattern with prefix pattern <a> are found by the Finding-HUS (p, sd$_{(p)}$, r) procedure with the parameters p=<a> , r=1, and its projected sequences sd$_{(a)}$. The details of the Finding-HUS (p, sd$_{(p)}$, r) procedure executed on this example are described below.

PSTEP 1: The 2-length sequential patterns with prefix as <a> are shown in table 8.

PSTEP 2: In table 8, the asu of each pattern is calculated by using position vector.

PSTEP 3: In table 8 , the asu of pattern <ae> is greater than the λ(=65) so it is put it in the set of HUS$_2$.

PSTEP 4: The patterns in table 8 are sequentially processed.

PSTEP 5: The procedure <(aa) , sd(aa), 2) is called and high utility sequential patterns with prefix as <(aa)> are found . Similarly all the patterns in table 8 are processed and high utility sequential patterns are found.

Table 8: Patterns generated with prefix as <a>

PATTERN	POSITION VECTOR	ASU
<aa>	< 1: {1, 2}, 3: {3,2} >	24
<(ad)>	<1: { 2,1} >	12
<ad>	<1: {1,1}, 5:{3,1} >	24
<ae>	<1: { 2,1}, 3:{3,2} ,4: {3,2},6:{2,2} >	65
<af>	<1: {2,2}, 4: {3,3} , 5:{3,1}>	36
<ab>	<2: {1,1}>	13
<ac>	<2:{1,15}, 3: {3,8}, 4:{3,2}, 6:{2,4}>	56
<(ac)>	<3: {2,8}>	14
<(ab)>	<4: {3,2}>	29

When new sequences are added to the original database, the indexing table is modified by following steps:

1) Add the corresponding position vector of the item of newly inserted sequence ,
2) Update actual sequence utility,

Suppose 2 new sequences say s$_8$ and s$_9$ of table 9 shown below are added to the existing database of table 1 then modification in the indexing strategy is done which is shown in table 10.

Table 9 :Newly Inserted Sequences

SID	SEQUENCES
S$_8$	<a(2),{b(1),e(4)},c(10),f(1)>
S$_9$	<a(1),b(2),c(3),d(1)

Table 10 : Modification in Indexing Strategy

ITEM	POSITION VECTOR	ASU
<a>	{8:2} {9:1}	42+6+3 =51
<b>	{8:1}{9:2}	30+10+20 =60
<c>	{8:10} {9:3}	31+10+3=44
<d>	{9:1}	12+6=18
<e>	{8:4}	40+20=60
<f>	{8:1}	12+2=14

Table 11: Generated Patterns

PATTERN	POSITION VECTOR	ASU
<aa>		9
<(ad)>		12
<ad>	<9:{1,1}>	9+9=18
<ae>	<8:{2,4}>	124+26=150
<af>	<8:{2,1}>	36+8=44
<ab>	<8: {2,1},9:{1,2}>	13+16+23=52
<ac>	<8: {2,10},9:{2,10}>	56+16+16=88
<(ac)>		14
<(ab)		19

Similarly other patterns can be generated by modified the values in the index table.

5. EXPERIMENTAL EVALUATION

A series of experiments were conducted to compare the performance of the proposed Incremental algorithm and PHUS algorithm with different parameter values. All the algorithms were implemented in Netbeans IDE 8.0.2 (JDK 1.8).

In these experiments, the sequence data was generated using java random number generator. In which the quantity of each item is ranged among 1 to 5 and utility of each item is ranged among 1 to 15.

The table 12 below shows the comparison of time required in mining HUS of existing and proposed algorithm on different value of threshold. Fig. 1 shows the comparison of time taken existing (PHUS) algorithm and proposed algorithm for different value of threshold.

Table 12 :Comparison of Time in mining HUS on 10000 records

Threshold (in %)	Time t$_1$ sec required in mining 8000 records using existing algorithm	Time t$_2$ sec required in mining 2000 records	t$_3$ = t$_1$+ t$_2$	Time t$_4$ sec required in mining 8000 records using proposed algorithm	Time t$_5$ sec required in mining 2000 records using proposed algorithm	t$_6$ = t$_4$+ t$_5$ (in sec)
0.01	109	136	245	102	16	118
0.008	154	192	346	152	22	174
0.006	204	255	459	201	31	232
0.004	269	336	605	263	56	319

Figure 1: Comparision of Time for mining HUS on 10000 records

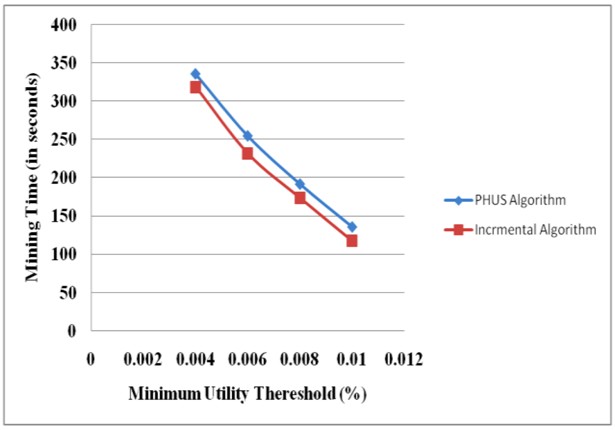
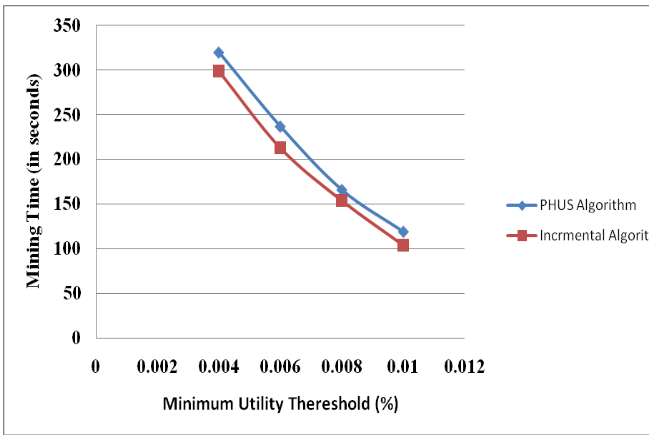


Table 13 :Comparison of Time in mining HUS on 8000 records

Threshold (in %)	Time t_1 sec required in mining 8000 records using existing algorithm	Time t_2 sec required in mining 2000 records using existing algorithm	$t_3 = t_1 + t_2$	Time t_4 sec required in mining 8000 records using proposed algorithm	Time t_5 sec required in mining 2000 records using proposed algorithm	$t_6 = t_4 + t_5$ (in sec)
0.01	109	136	245	102	16	118
0.008	154	192	346	152	22	174
0.006	204	255	459	201	31	232
0.004	269	336	605	263	56	319

Figure 2: Comparision of Time for mining HUS on 8000 records



7. Conclusion

In this paper, an incremental algorithm for mining high utility sequential pattern from incremental sequential database was proposed. By using this approach the proposed algorithm do not need to rescan the existing data for incremental sequences. An effective indexing structure helps to quickly find the relevant quantitative sequences for prefixes to be processed in the recursive process and also helps in reducing the calculation of actual sequence utility. Finally, the experimental results show that proposed algorithm executes faster for incremental sequences.

In future, an attempt to further apply the principle of high utility sequential pattern mining to other practical applications is such as by the problem of mining high utility sequential pattern in data stream environments.

6. REFERENCES

[1] J. Pei, J. Han, B. Mortazavi-As1, H. Pinto, Q. Chen , U. Dayal, M.Hsu ,"Prefixspan,: Mining Sequential Patterns Efficiently by Prefix-Projected Pattern Growth", In Natural Sciences and Engineering Research Council of Canada (grant NSERC-A3723), the Net works of Centres of Excellence of Canada (grant NCWIRIS-3). and the Hewlett-Packard Lab, U.S.A. , pp. 215-220. IEEE,2001.

[2] C. Lina,T. Hong, W. Lua, "An effective tree structure for mining high utility itemsets" Expert Systems with Applications,vol.38(6),PP.7419-7424. ELSEVIER,2010.

[3] T. P. Hong, C.H. Lee, and S.L. Wang,"Mining high average-utility itemsets," In Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on, pp. 2526-2530. IEEE, 2009.

[4] S. Shankar, T. Purusothaman, S. Jayanthi, and N. Babu , "A Fast Algorithm for Mining High Utility Itemsets," In Advance Computing Conference, 2009. IACC 2009. IEEE International, pp. 1459-1464. IEEE, 2009.

[5] C. Lin, T. Hong , G. Lan, H. Chen, H. Kao "Incrementally Mining High Utility Itemsets in Dynamic Database", In IEEE International Conference on Granular Computing PP. 303-307.IEEE 2010

[6] J. Yin, Z. Zheng, and L. Cao, "USpan: An efficient algorithm for mining high utility sequential patterns, "In Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 660-668. ACM, 2012.

[7] G. Lan , T. Hong , V. Tseng , S. Wang , "An Improved Approach for Sequential Utility Pattern Mining" , In 2012 IEEE International Conference on Granular Computing PP. 226-230. IEEE,2012.

[8] G. Lan ,T. Hong , V. Tseng , S. Wang , "Applying maximum utility measure in high utility Sequential Pattern", Expert Systems with Applications,vol.41 pp. 5071-5081. ELSEVIER, 2014.

[9] D.Parmar , Y.Rathod , M.Patel , "Survey on High Utility oriented Sequential Pattern Mining ", IEEE International Conference on computational Intelligence and and computing Research ,IEEE, 2013.

[10] S. Londhe1, R. Mahajan2, B. Bhoyar3 , "Overview on Methods for Mining High Utility Itemset from Transactional Database "International Journal of Scientific Engineering and Research (IJSER) , Volume 1 Issue 4, pp.-15-19, IJSER , 2013.

[11] S. Bhattacharya1, D. Dubey2, “High Utility Itemset Mining” , In International Journal of Emerging Technology and Advanced Engineering , Volume 2, Issue 8, pp.476-481. IJETAE ,2012

[12] Vinutha C, Yogish H.K , “Survey On: Mining High Utility Itemsets from Transactional Database “, In IOSR Journal of Computer Engineering (IOSR-JCE), Volume 16, Issue 2, pp.-91-93, IOSR , 2014.

[13] C. Chand, A. Thakkar, A. Ganatra, ““Sequential Pattern Mining: Survey and Current Research Challenges”, International Journal of Soft Computing and Engineering (IJSCE) , Volume-2, Issue-1. pp-185-193, IJSCE, 2012.

[14] Mabroukeh.n.r and Ezeife.c.i, 2010 “A Taxonomy of Sequential Pattern Mining Algorithms, “ACM Computing Surveys, Vol. 43, No. 1, Article 3,pp. 3-1 to 3-43, ACM.

[15] S.Vijayarani , S.Deepa, “Sequential Pattern Mining – A Study”, International Conference on Research Trends in Computer Technologies (ICRTCT),Proceedings published in International Journal of Computer Applications® (IJCA) pp.14-18. ICRTCT-2013.

[16] C. Mooney , J. Roddick “Sequential Pattern Mining – Approaches and Algorithms “ Vol. V, No. N, M 20YY, ACM , pp. 1–46.

[17] J. Pillai , O.P.Vyas, “Overview of Itemset Utility Mining and its Applications ” International Journal of Computer Applications Volume 5– No.11, pp. 9-13. IJCA -2010.