SIMULATED ANALYSIS OF EFFICIENT ALGORITHMS FOR MINING TOP-K HIGH UTILITY ITEMSETS

Surbhi Choudhary[1], Devendra Nagal[2], Swati Sharma[3]
[1]PhD Research Scholar, Dept. of Computer Applications, JNU Jodhpur
[2,3]Faculty, Dept. of Electrical Engineering, JNU Jodhpur

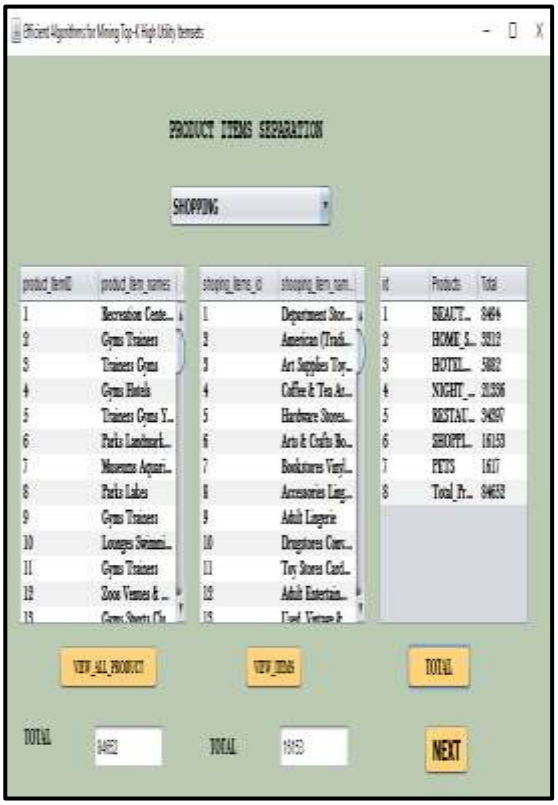
***Abstract*— High utility sequential pattern mining is an emerging topic in the data mining community. Compared to the classic frequent sequence mining, the utility framework provides more informative and actionable knowledge since the utility of a sequence indicates business value and impact. However, the introduction of "utility" makes the problem fundamentally different from the frequency-based pattern mining framework and brings about dramatic challenges. Although the existing high utility sequential pattern mining algorithms can discover all the patterns satisfying a given minimum utility, it is often difficult for users to set a proper minimum utility. A too small value may produce thousands of patterns, whereas a too big one may lead to no findings. In this paper, we propose 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 designed to identify top-k high utility sequential patterns without minimum utility. In addition, three effective features are introduced to handle the efficiency problem, including two strategies for raising the threshold and one pruning for filtering unpromising items. Our experiments are conducted on both synthetic and real datasets..**

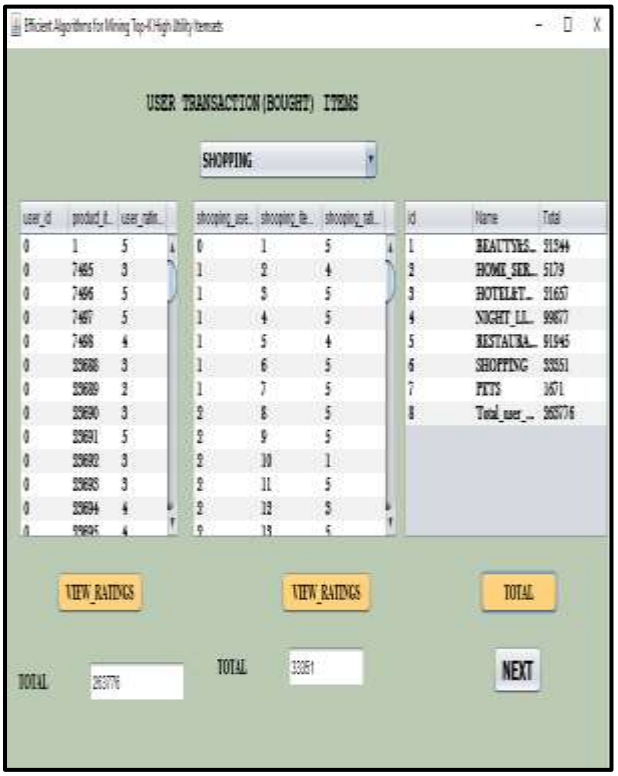
I. INTRODUCTION

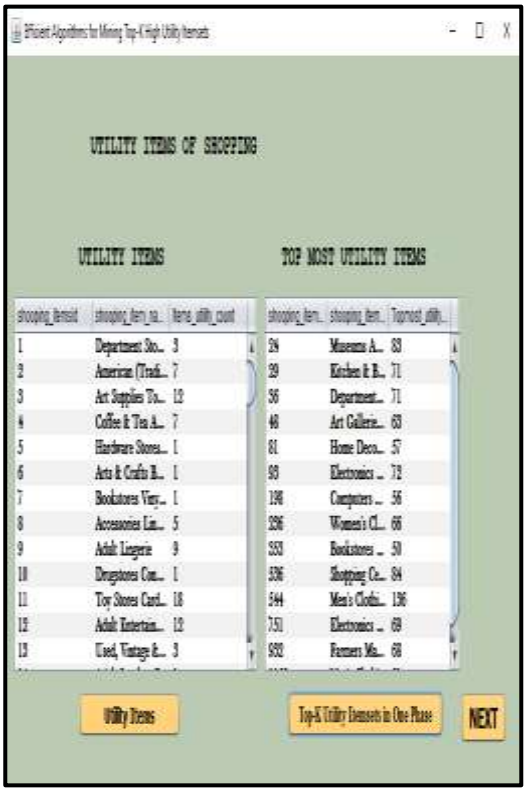
FREQUENT sequential pattern mining [1], as one of thefundamental research topics in data mining, discovers frequent subsequences in sequence databases. It is very usefulfor handling order-based business problems, and has beensuccessfully adopted to various domains and applicationssuch as complex behaviour analysis [1] and gene sequenceanalysis [2], [3], [4]. In the frequency-based framework fortypical sequence analysis, the downward closure property(also known as Apriori property) [1] plays a fundamentalrole in identifying frequent sequential patterns.However, taking the frequency to measure pattern in-terestingness may be insufficient for selecting actionablesequences associated with expected quality and businessimpact, because the patterns identified under the frequency(support) framework do not disclose the business valueand impact. To solve the above problems, the conceptutility is introduced into sequential pattern mining to select sequences of high utility by considering the quality and value (such as profit) of itemsets. This leads to anemerging area, high utility pattern mining [1], [2], [3], [8],[9] and high utility sequential pattern mining, which selects interesting patterns / sequentialpatterns based on minimum utility rather than minimumsupport. The utility-based patterns are proven to be moreinformative and actionable for decision-making than the frequency-based ones [2]. For instance, in [4], [5], the authors discuss the extraction of profitable behaviours fromthe mobile commerce environments. [4] Proposes methodsto mine high utility sequences from web logs by assigningeach page an impact/significance. In [6], a US pan algorithmis built for utility-based sequential pattern mining satisfyinga predefined minimum utility.Although algorithms such as USpan can obtain high utilitysequences based on a given minimum utility, it is verydifficult for users to specify an appropriate minimum utilitythreshold and to directly obtain the most valuable patterns.This is because the complexity of utility-based sequencedatabases (which may be different from the classic itemsets),determining multiple factors including the distribution of theitems and utilities, density of the database, lengths of thesequences, and so on. Consequently, it is not surprising that,with a same minimum utility threshold, some datasets mayproduce millions of patterns while others may contributenothing. The challenge here is that it may not be doableto tune the threshold to capture the expected number ofpatterns. This is because the sensitivity of the thresholdmakes it hard to tune for a variety of databases. It maybe very costly and time consuming to achieve the properthreshold for the desired patterns.In fact, the classic frequency/support based pattern miningalso faces the same challenge. Accordingly, the concept ofextracting top-k patterns has been proposed in [2], [4], [7],[8] to select the patterns with the highest frequency. Inthe top-k frequent pattern mining, instead of letting a userspecify a threshold, the top-k pattern selection algorithmsallow a user to set the number of top-k high frequencypatterns to be discovered. This makes it much easier andmore intuitive and practical than determining a minimumsupport; also the determination of k by a user is morestraightforward than considering data characteristics, whichare often invisible to users, for choosing a proper threshold. The easiness for users to determine k does not indicatethe simplicity of developing an efficient algorithm for selecting top-k high utility sequential patterns. In the utilityframework, TKU is the only method for mining top-k high utility itemsets, to the best of our knowledge. Nowork is reported on mining top-k high utility sequences. There is significant difference between top-k utility itemsetmining and top-k utility sequence mining in which the orderbetween itemsets is considered. In fact, the problem of top-khigh utility sequence mining is much more challenging thanmining top-k high utility itemsets. First, as

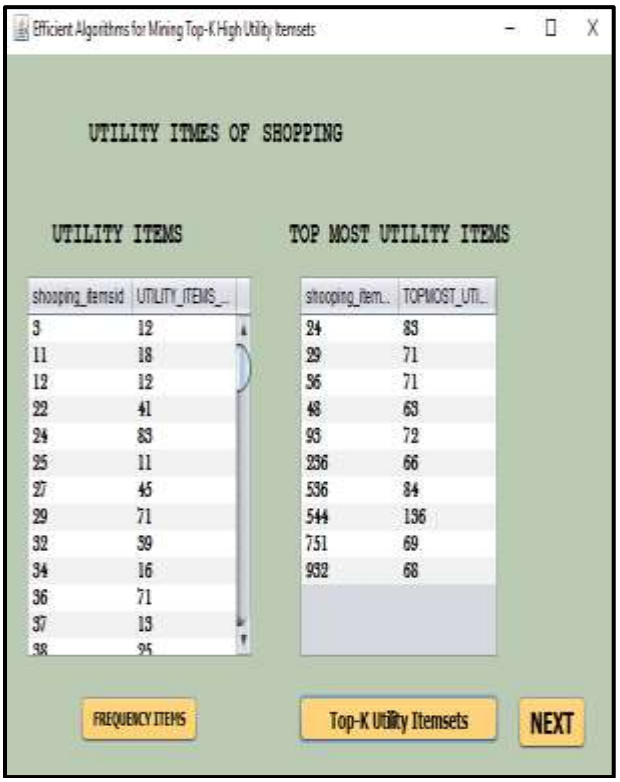
with high utilityItemset mining, the downward closure property does not holdin the utility-based sequence mining. This means that the existing top-k frequent sequential pattern mining algorithms [7]cannot be directly applied. Second, compared to top-k highutility itemset mining [8], utility-based sequence analysisfaces the critical combinational explosion and computationalcomplexity caused by sequencing between itemsets. Thismeans that the techniques in [9] cannot be directly transferred to top-k high utility sequential pattern mining either.Third, since the minimum utility is not given in advance, thealgorithm essentially starts the searching from 0 minimumsupports. This not only incurs very high computational costs,but also the challenge of how to raise the minimum thresholdwithout missing any top-k high utility sequences.

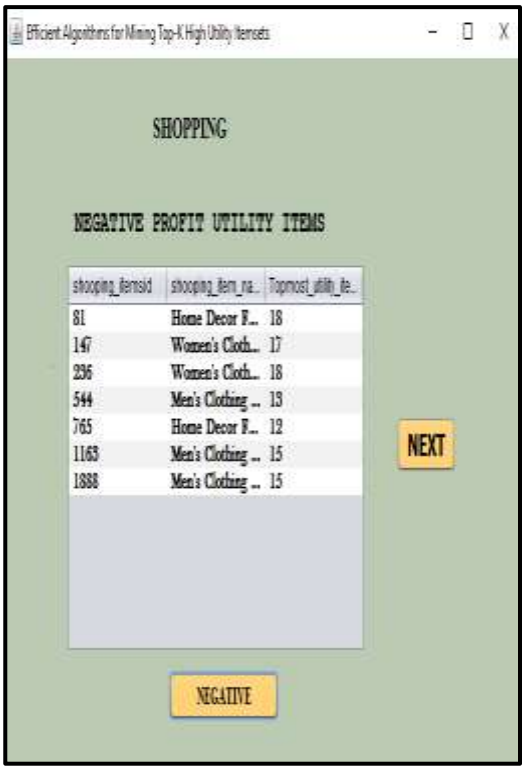
II. Simulation Results



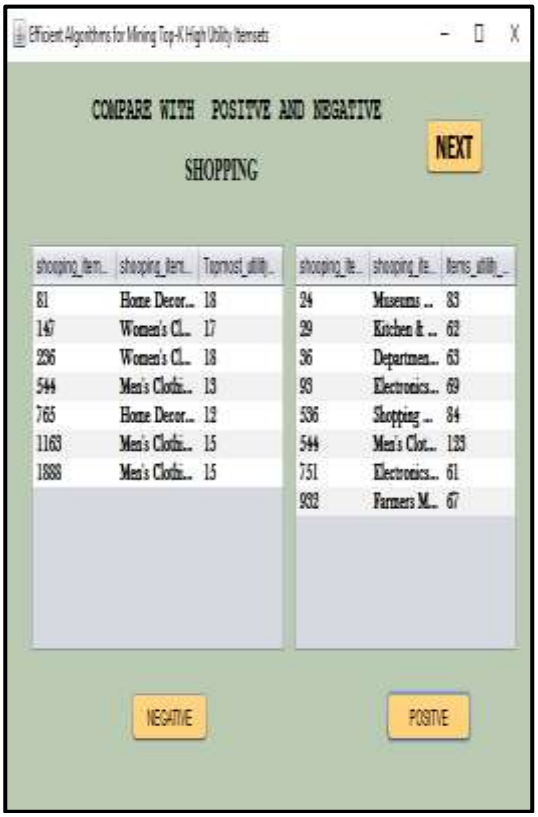

USER TRANSACTION (BOUGHT) ITEMS
SHOPPING
VIEW_RATINGS
VIEW_RATINGS
TOTAL
TOTAL
TOTAL
NEXT

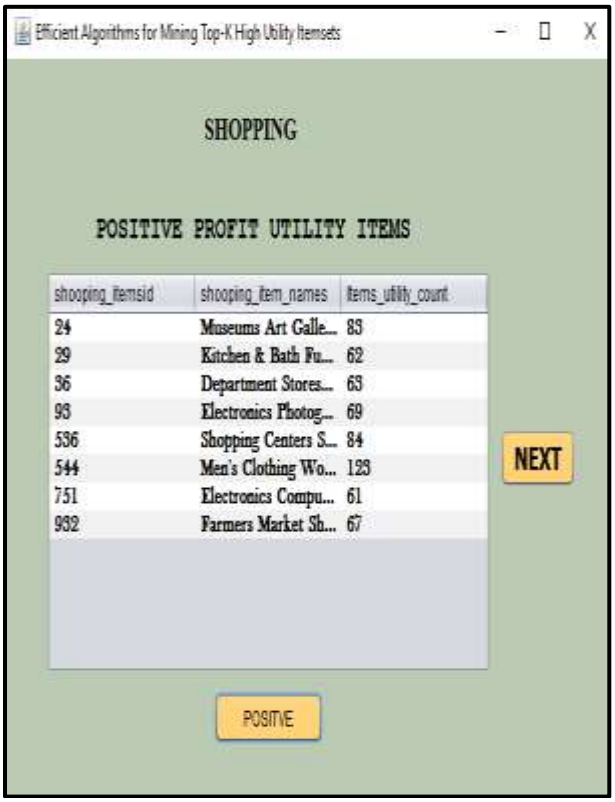

UTILITY ITEMS OF SHOPPING
UTILITY ITEMS
TOP MOST UTILITY ITEMS
Utility Items
Top-K Utility Itemsets in One Phase
NEXT

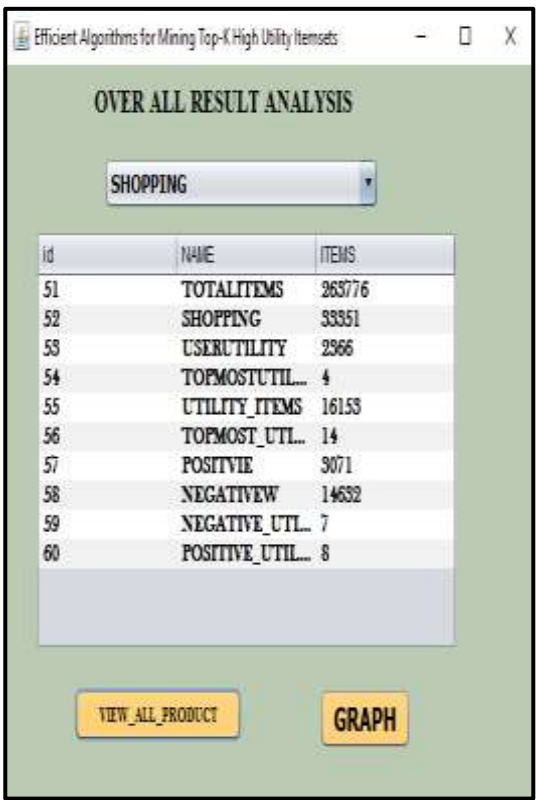

UTILITY ITMES OF SHOPPING
UTILITY ITEMS
TOP MOST UTILITY ITEMS
FREQUENCY ITEMS
Top-K Utility Itemsets
NEXT

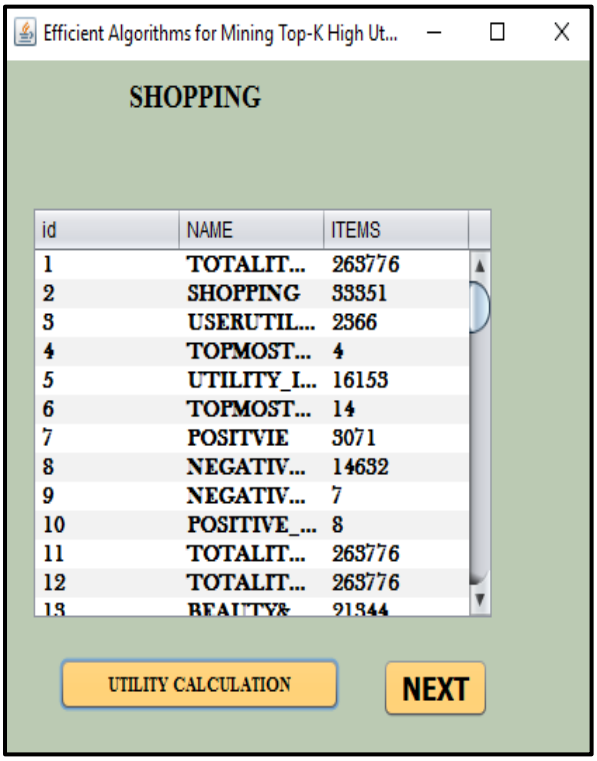

SHOPPING
NEGATIVE PROFIT UTILITY ITEMS
NEXT
NEGATIVE


Efficient Algorithms for Mining Top-K High Utility Itemsets
SHOPPING
NEGATIVE UTILITY ITEMS
POSITIVE UTILITY ITEMS
NEGATIVE
POSITIVE
NEXT

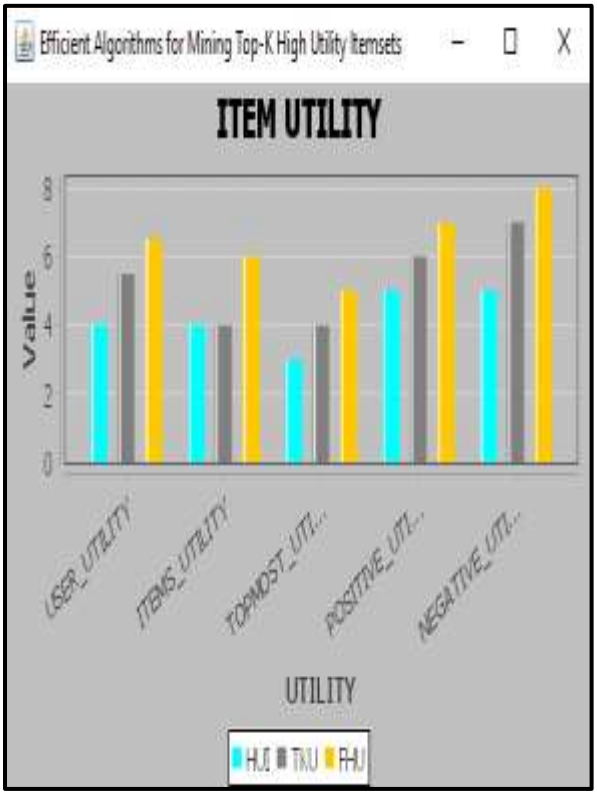

Efficient Algorithms for Mining Top-K High Utility Itemsets
COMPARE WITH POSITVE AND NEGATIVE
SHOPPING
NEXT
NEGATIVE
POSITIVE

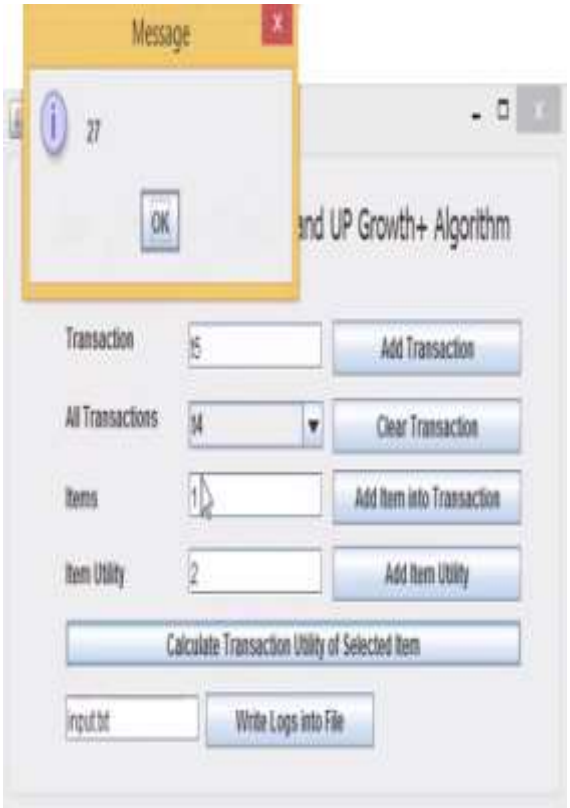

Efficient Algorithms for Mining Top-K High Utility Itemsets
SHOPPING
POSITIVE PROFIT UTILITY ITEMS
shopping_itemsid
shopping_item_names
items_utility_count
24 Museums Art Galle... 83
29 Kitchen & Bath Fu... 62
36 Department Stores... 63
93 Electronics Photog... 69
536 Shopping Centers S... 84
544 Men's Clothing Wo... 123
751 Electronics Compu... 61
932 Farmers Market Sh... 67
NEXT
POSITIVE

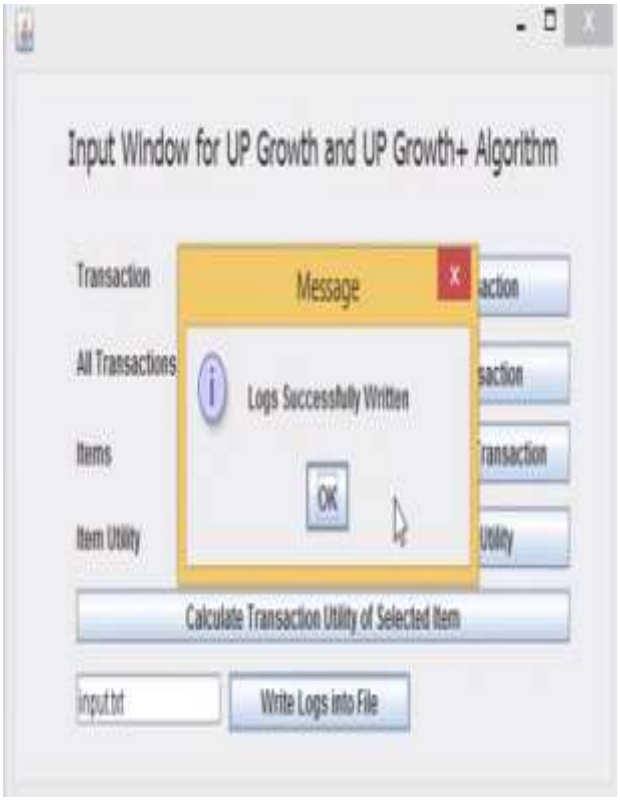

Efficient Algorithms for Mining Top-K High Utility Itemsets
OVER ALL RESULT ANALYSIS
SHOPPING
id
NAME
ITEMS
51 TOTALITEMS 265776
52 SHOPPING 33351
53 USERUTILITY 2366
54 TOPMOSTUTIL... 4
55 UTILITY_ITEMS 16153
56 TOPMOST_UTL... 14
57 POSITVIE 3071
58 NEGATIVEW 14632
59 NEGATIVE_UTL... 7
60 POSITIVE_UTIL... 8
VIEW_ALL_PRODUCT
GRAPH

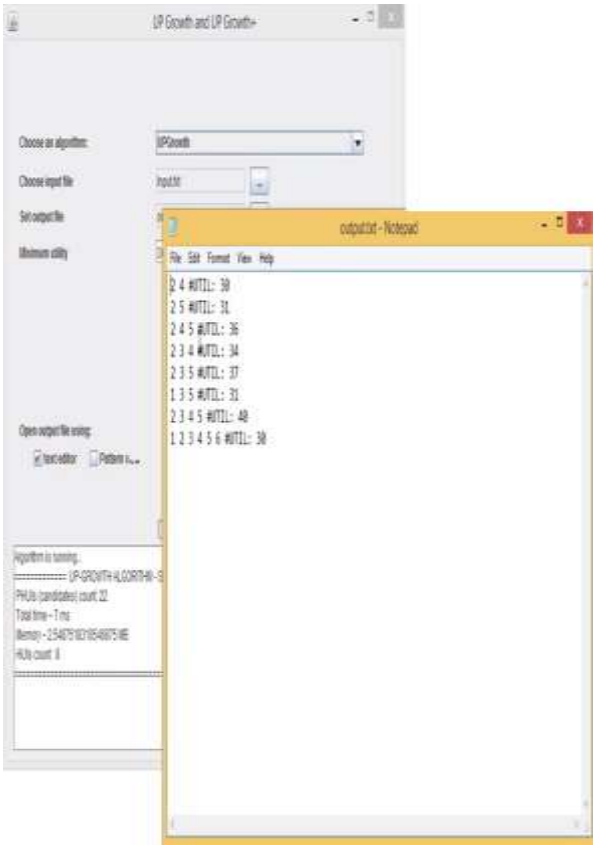

Efficient Algorithms for Mining Top-K High Ut...
SHOPPING
id
NAME
ITEMS
1 TOTALIT... 263776
2 SHOPPING 33351
3 USERUTIL... 2366
4 TOPMOST... 4
5 UTILITY_I... 16153
6 TOPMOST... 14
7 POSITVIE 3071
8 NEGATIV... 14632
9 NEGATIV... 7
10 POSITIVE_... 8
11 TOTALIT... 263776
12 TOTALIT... 263776
UTILITY CALCULATION
NEXT

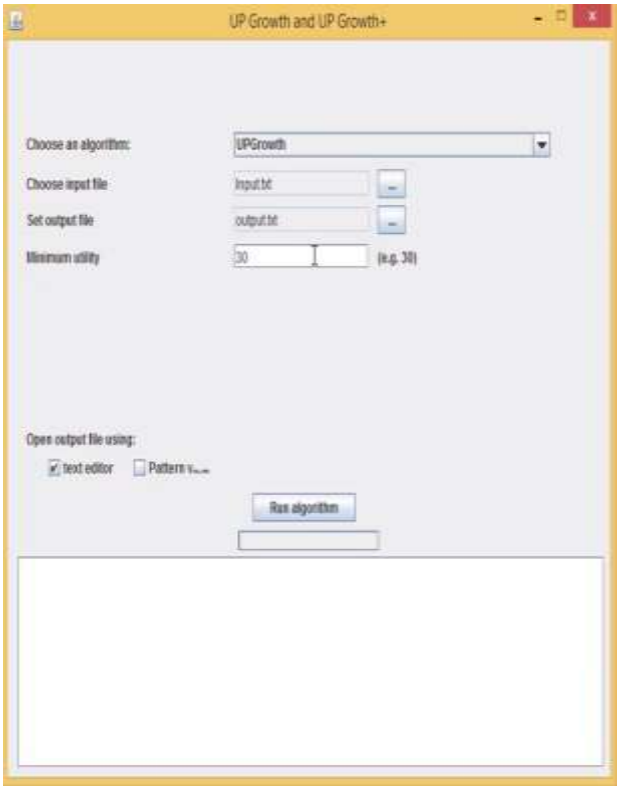

Input Window for UP Growth and UP Growth+ Algorithm
Transaction
Add Transaction
All Transactions
Clear Transaction
Items
Add Item into Transaction
Item Utility
Add Item Utility
Calculate Transaction Utility of Selected Item
input.txt
Write Logs into File

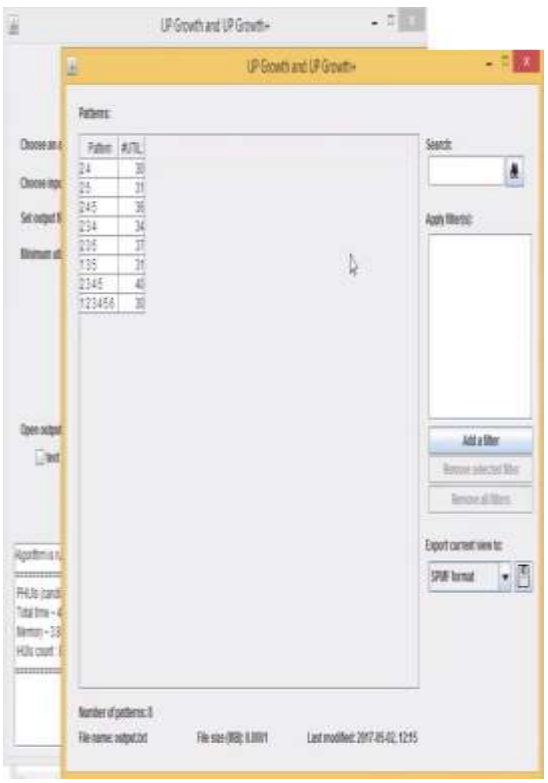

Efficient Algorithms for Mining Top-K High Utility Itemsets
ITEM UTILITY
Value
USER_UTILITY
ITEMS_UTILITY
TOPMOST_UTI...
POSITIVE_UTI...
NEGATIVE_UTI...
UTILITY
HUI
TKU
PHU

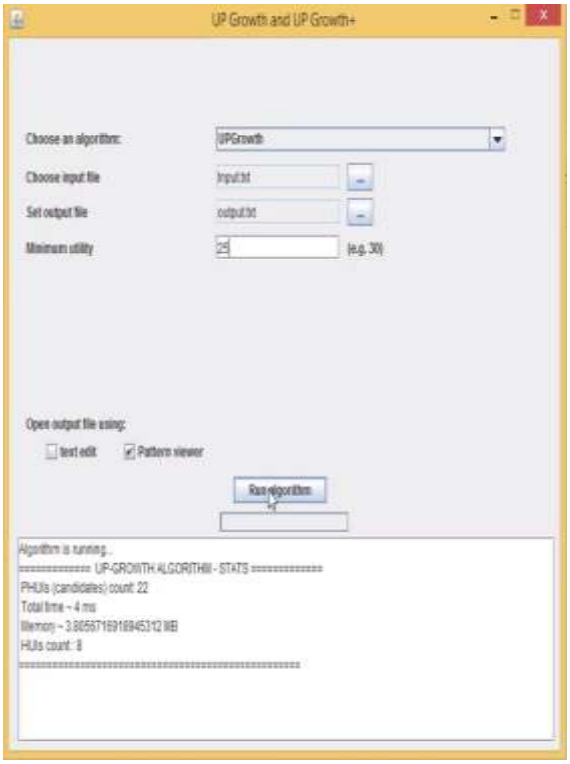

Message
27
OK
Transaction
Add Transaction
All Transactions
Clear Transaction
Items
Add Item into Transaction
Item Utility
Add Item Utility
Calculate Transaction Utility of Selected Item
input.txt
Write Logs into File

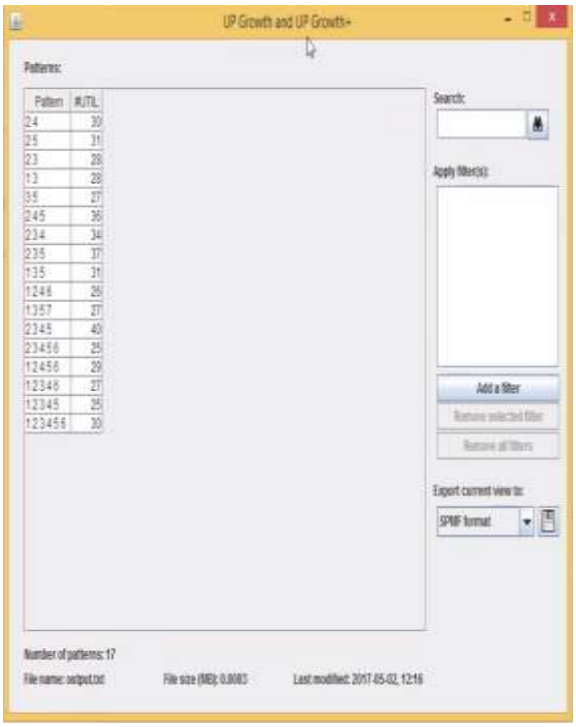

Input Window for UP Growth and UP Growth+ Algorithm
Transaction
All Transactions
Items
Item Utility
Message
Logs Successfully Written
OK
Calculate Transaction Utility of Selected Item
input.txt
Write Logs into File


UP Growth and UP Growth+
Choose an algorithm:
UPGrowth
Choose input file
Set output file
Minimum utility
output.txt - Notepad
File Edit Format View Help
2 4 #UTIL: 30
2 5 #UTIL: 31
2 4 5 #UTIL: 36
2 3 4 #UTIL: 34
2 3 5 #UTIL: 37
1 3 5 #UTIL: 31
2 3 4 5 #UTIL: 40
1 2 3 4 5 6 #UTIL: 30


UP Growth and UP Growth+
Choose an algorithm:
UPGrowth
Choose input file
input.txt
Set output file
output.txt
Minimum utility
30
(e.g. 30)
Open output file using:
text editor
Pattern
Run algorithm


UP Growth and UP Growth+
Patterns:
Pattern
#UTIL
Search:
Add a filter
Export current view to:
SPMF format





III. Comparison with UPGrowth and UPGrowth+ Algorithm

In this project, two algorithms utility pattern growth (UP-Growth) and UP-Growth+, for mining high utility itemsets are used to compare. Performance of UP-Growth and UP-Growth+ become more efficient since database contain long transactions and generate fewer number of candidates than FP-Growth.

Two important problems are always in consideration first is how minimize number no of candidates and another is how to remove space and time complexity. Also, choosing an appropriate minimum utility threshold is a difficult task for application users: if the threshold is high, there might be no HUI; if the threshold is low, there might result too many HUIs, and the mining performance might be severely affected, even leading to memory overflow. It would also be a time-consuming task if one tries to determine the threshold value through various testing calculations.

To address this issue, Wu [10] proposes top-k algorithm, mining the top k itemsets with the highest utility values without presetting the minimum threshold.

IV. Conclusion

In this paper, we have studied the problem of top-k high utility item sets mining, where k is the desired number of high utility item sets to be mined. 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. TKU is the first two-phase algorithm for mining top-k high utility item sets, which incorporates five strategies PE, NU, MD, MC and SE to effectively raise the border minimum utility thresholds and further prune the search space. On the other hand, TKO is the first one-phase algorithm developed for top-k HUI mining, which integrates the novel strategies RUC, RUZ and EPB to greatly improve its performance. Empirical evaluations on different types of real and synthetic datasets show that the proposed algorithms have good scalability on large datasets and the performance of the proposed algorithms is close to the optimal case of the state-of-theart two-phase and one-phase utility mining algorithms. Although we have proposed a new framework for top-k HUI mining, it has not yet been incorporated with other utility mining tasks to discover different types of top-k high utility patterns such as top-k high utility episodes, top-k closed high utility item sets, top-k high utility web access patterns and top-k mobile high utility sequential patterns. These leave wide rooms for exploration as future work.

References

[1] R. Agrawal and R. Srikant, “Fast algorithms for mining association rules,” in Proc. Int. Conf. Very Large Data Bases, 1994, pp. 487–499.
[2] 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.
[3] 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.
[4] R. Chan, Q. Yang, and Y. Shen, “Mining high-utility itemsets,” in Proc. IEEE Int. Conf. Data Mining, 2003, pp. 19–26.
[5] P. Fournier-Viger and V. S. Tseng, “Mining top-k sequential rules,” in Proc. Int. Conf. Adv. Data Mining Appl., 2011, pp. 180–194.

[6] P. Fournier-Viger, C.Wu, and V. S. Tseng, "Mining top-k association rules," in Proc. Int. Conf. Can. Conf. Adv. Artif. Intell., 2012, pp. 61–73.

[7] P. Fournier-Viger, C. Wu, and V. S. Tseng, "Novel concise representations of high utility item sets using generator patterns," in Proc. Int. Conf. Adv. Data Mining Appl. Lecture Notes Comput. Sci., 2014, vol. 8933, pp. 30–43.

[8] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," in Proc. ACM SIGMOD Int. Conf. Manag. Data, 2000, pp. 1–12.

[9] J. Han, J. Wang, Y. Lu, and P. Tzvetkov, "Mining top-k frequent closed patterns without minimum support," in Proc. IEEE Int. Conf. Data Mining, 2002, pp. 211–218.

[10] S. Krishnamoorthy, "Pruning strategies for mining high utility itemsets," Expert Syst. Appl., vol. 42, no. 5, pp. 2371–2381, 2015.

[11] C. Lin, T. Hong, G. Lan, J. Wong, and W. Lin, "Efficient updating of discovered high-utility itemsets for transaction deletion in dynamic databases," Adv. Eng. Informat., vol. 29, no. 1, pp. 16–27, 2015.

[12] G. Lan, T. Hong, V. S. Tseng, and S. Wang, "Applying the maximum utility measure in high utility sequential pattern mining," Expert Syst. Appl., vol. 41, no. 11, pp. 5071–5081, 2014.

[13] Y. Liu, W. Liao, and A. Choudhary, "A fast high utility itemsets mining algorithm," in Proc. Utility-Based Data Mining Workshop, 2005, pp. 90–99.

[14] M. Liu and J. Qu, "Mining high utility itemsets without candidate generation," in Proc. ACM Int. Conf. Inf. Knowl. Manag., 2012, pp. 55–64.