Chapter 8

Conclusion and Future Directions

8.1 Conclusions

High utility itemsets (HUIs) mining is a subfield of frequent itemsets mining (FIM) which has wide application areas. The traditional FIM focuses on extracting patterns (or itemsets) based on frequency/support framework which does not have business value and thus is not actionable for business decision-making. Many companies are looking for better ways to gain value from their data to be competitive in the market. Traditional FIM may not fulfill the requirement of the corporate world because FIM does not consider the quantity and importance (unit profit, price, risk, and cost, etc.). Importance based mining play an important role in the field of itemsets mining. Therefore, HUIs mining has been introduced that includes quantity and importance of items. The main objective of HUIs mining is to extract valuable and useful itemsets by considering a business objective such as profit, user's interest, cost, etc. HUIs mining not only brought valuable information but also new problems and challenges. The key challenges in HUIs mining are as follows.

First, HUIs mining does not follow *downward closure property* because the utility of an itemset may be smaller, equal or greater to the utility of its supersets (or subsets). Proposing a new strategy which follows *downward closure property* is a big challenge in HUIs mining. Second, HUIs mining algorithms store more information. Hence, the major challenge in HUIs mining to propose an efficient strategy to prune the search space. Third, traditional HUIs mining algorithms only consider positive utility of items. However, in real-life negative utility has found everywhere and has a wide range of applications. Handling negative utility is also a major challenge. Fourth and last challenge addressed in this thesis, HUIs mining suffers from a large amount of very small and redundant itemsets. The challenge here is to explore approaches and algorithms which can efficiently and effectively summarize the itemsets without losing or losing only the smallest amount of information. In this thesis work, we propose five solutions to HUIs mining to target the above-discussed challenges. First two works (EHIL and TKEH) only consider positive utility value. Rest three works (EHIN, EHNL and CHN) consider both positive and negative utility value.

In Chapter 3, we propose an efficient algorithm named EHIL which overcomes the limitation of mining lots of very small itemsets and very long itemsets. The proposed algorithm mines the HUIs considering length constraints. EHIL propose many definitions and properties. It redefines sub-tree and TWU pruning strategies by incorporate length constraints. EHIL uses array-based utility counting technique that calculates the utility in linear time and consumes negligible memory space. The experimental results show that proposed algorithm highly reduced the execution time and memory requirements. EHIL algorithm outperforms the state-of-the-art algorithm FHM+ for both in runtime and memory aspect for all our observations. The experimental results show that EHIL is relatively up to 1896 times faster in execution time than FHM+ algorithm. In memory comparison, EHIL uses up to 28 times less memory than FHM+ algorithm.

In Chapter 4, we propose an efficient top-k HUIs mining algorithm named TKEH. TKEH utilize three strategies (RIU, CUD and COV) to raise internal minimum utility threshold. To show the effects of all the techniques and strategies, five versions of TKEH are proposed separately named as TKEH, TKEH(CUD), TKEH(RIU), TKEH(sup) and TKEH(tm). The experimental results show that proposed algorithms outperform the state-of-the-art algorithm kHMC. The scalability analysis shows that TKEH is highly scalable for dense and sparse datasets. The runtime improvement analysis shows that the proposed algorithm up to three orders of magnitude faster than kHMC. Moreover, the proposed algorithm is always memory efficient for dense datasets.

In Chapter 5, we propose an algorithm named EHIN. EHIN proposes several properties to handle negative utility values. EHIN also proposes two new upper bounds to prune the search space. The experimental results on nine datasets show that proposed algorithm outperforms the state-of-the-art algorithm FHN for both in runtime and memory aspect in all our observations. The runtime gap between the proposed algorithm and FHN become larger as we set lower min_util which indicates that proposed algorithms can run for more lower min_util threshold than FHN. The experimental results show that EHIN algorithm is relatively 28 times faster in execution time and consumes up to 10 times less memory than FHN algorithm.

In Chapter 6, we propose an algorithm named EHNL which mines HUIs with negative utility value and length constraints. EHNL incorporates length constraints in negative utility that removes very small and very large itemsets. In order to achieve the space and time efficiency, EHNL proposes redefined sub-tree pruning strategy to reduce the search space. EHNL presents several properties and definitions to tackle with negative utility and length constraints. To check the efficiency of proposed techniques and strategies, we propose two versions of EHNL named EHNL(RSUP) and EHNL(TM). The experimental results show that the proposed algorithms mine HUIs efficiently with real-life or benchmark datasets.

In Chapter 7, we propose a closed HUIs mining algorithm named CHN which mines non-redundant itemsets. CHN proposes two pruning to remove non-HUIs. CHN also proposes Bi-directional extension closure checking technique to speed up the mining process. It also proposes two strategies to prune the non-closed HUIs by using bi-directional closure property. The experimental results on dense and sparse datasets show that the proposed algorithm is efficiently mine the closed HUIs. CHN is relatively up to 44 times faster in execution time than the state-of-the-art algorithm FHN. CHN consumes up to 13 times less memory than FHN.

In this thesis, we have focused on HUIs mining. We have proposed five algorithms. Two of these five algorithms mine HUIs with positive utility only. Rest three algorithms mine HUIs with positive and negative utility. All the proposed algorithms are compared with the state-of-the-art algorithms given in the sub-domain. The results generated by our algorithms are compared with the results of the state-of-the-art algorithms. Through exhaustive experiments, we have proved that our algorithms perform better than the state-of-the-art algorithms in comparison with execution time and memory usage.

8.2 Future Directions

There are several future directions for the work this thesis presents.

8.2.1 Mining High Utility Itemsets with Positive Utility only

1. **High Utility Itemsets Mining Considering Length Constraints:** Constraint-based mining plays an important role to fulfill requirements of end user. Length-based constraint mining can be implemented easily and solve the problem of generating very small and very large itemsets. Length constraint-based mining concepts can be applied in many areas such as HUIs mining from data stream, incremental dataset, on-shelf and sequential datasets, etc.

2. **Top-k High Utility Itemsets Mining:** Propose a new internal minimum utility threshold is always required to improve the mining process. Most of the state-of-art algorithms work with static datasets. If the dataset is updated, static type of algorithms need to be run again to obtain the updated rules. This type of approach is not efficient because sometimes only small changes are made to a datasets and algorithms have to scan whole dataset again to mine rules. A solution to this problem is to design a top-k HUIs mining algorithm with dynamic datasets. To the best of our knowledge, T-HUDS[14] is the only algorithm to mine top-k HUIs from data stream. T-HUDS is two-phase algorithm. Hence, there is lots of scopes to propose efficient algorithms.

   Some other extensions of top-k HUIs which mine rich itemsets in various ways, such as mining itemsets from uncertain data, top-k high utility sequential itemsets mining, periodic top-k HUIs mining and top-k on-shelf HUIs mining, etc.

8.2.2 Mining High Utility Itemsets with Positive and Negative Utility

3. **High Utility Itemsets Mining with Negative Utility Value:** Mining HUIs with negative utility value is a new area to explore. A very few works have been proposed to mine HUIs with negative utility value for transactional [29, 30], data stream [68], on-shelf [67, 57], sequential [70] and uncertain [75] datasets. Therefore, there are lots of research to do.

4. **High Utility Itemsets Mining with Negative Utility Value and Length Constraints:** Constraint-based mining with negative utility requires to mine more relevant rule according to the need of users. In literature, no algorithm has been found to mine HUIs with negative utility and constraints. Therefore, in this field, lots of research is required.

5. **Closed High Utility Itemsets Mining with Negative Utility Value:** Recently many closed algorithms for HUIs are proposed [65, 58, 61]. However, till now no any algorithm have been presented for closed HUIs with negative utility value. Therefore, closed HUIs with negative utility value to be a good area to explore. Closed HUIs mining with negative utility can be explored in some other interesting areas such as data stream, on-shelf, sequential and uncertain datasets, etc.

8.2.3 Other High Utility Itemsets Problems

The proposed ideas of constraint, closed and negative utility based mining can be utilized in some other interesting area such as up-to-date HUIs mining, high average utility itemsets mining, fuzzy HUIs mining, periodic HUIs mining, episode HUIs mining HUIs, mining using multiple minimum utility thresholds, etc. Furthermore, to reduce the execution time, the proposed ideas can be utilized in parallel and distributed HUIs mining.