A REVIEW ON VARIOUS HIGH UTILITY ITEMSET MINING ALGORITHMS

L. Anju, V. Sangeetha*

Abstract

Data mining techniques basically focusses on finding hidden information from huge volume of data. Pattern mining is one of the most interesting research areas in data mining that aims to mine different patterns. Those are useful for various real - world applications. The main types of patterns that are mined from the data can be clusters, itemset, outliers, sequential patterns, trends etc. This paper focuses on HUIM and a comparative study over the various algorithms used for HUIM for the last few years.

Keywords: Data Mining, Pattern-mining, High Utility Itemset Mining.

I INTRODUCTION

The information around us is continuously growing. Data mining can be defined as a non -trivial process of extracting knowledge which is hidden, previously unknown and potentially useful, from large data base [1]. The mined data provides some knowledge in discovering new patterns which could be used in real world applications. Data mining is considered as an inter-disciplinary field which provides techniques which has lot do to in information industry and with the society. The areas which are much benefitted by the patterns or knowledge discovered by data mining techniques are Business Transactions, Medical data, Personal data, Scientific data, Text reports etc.

II PATTERN MINING

Pattern mining is one of the important techniques that identifies the rules that will describe specific patterns like sequential patterns, outliers, itemset, clusters, graph structure, trends within the data. Pattern mining technique was first used for Market Basket Analysis whose purpose was to discover the items which is purchased by the customers frequently.

Several methods have been developed in the data mining area which could be used to extract interesting patterns. Among them most important and fundamental datamining techniques in various domain [2] are Frequent pattern mining (FPM) [3][4], Association rule mining (ARM) [5,6], Frequent episode mining (FEM)[7,8,9,10], Sequential Pattern Mining (SPM) [11]. The decision on which pattern mining technique is to be used is basically meant on the choice of measure for which are needed to find patterns. Among the various pattern mining, techniques utility mining is the emerging topic in the field of data mining.

III HIGH UTILITY PATTERN MINING

HUIM is an extensions of frequent pattern mining (FPM). It is one among different areas in which most research works are carried out. It has very important impact in various applications such as finance, e-commerce, social network, manufacturing etc. Utility pattern mining is focusing on mining the patterns with high value (like maximum profit) from large databases with the prime motto of discovering and analyzing high utility itemset which could help in improving the profit of the specified organization.

The criteria which are basically considered for high utility pattern mining are profit, weight, frequency etc. High utility itemset are measured as utility function in which it defines the criteria such as the amount of profit got from a

Department of Computer Science,
Karpagam Academy of Higher Education , Coimbatore, IndiaTamil Nadu
*Corresponding Author

sale, or the amount of time spent by a user on a webpage [12].

In HUIM, the input is the transactional database with item quantities and their weights. The weight represents the importance of the items. The main aim of this problem is to discover all itemset that will yield highest profit from the transactional data base.

In HUIM algorithms, two parameters are basically considered as input. Among which one is the transactional database and other is the minimum utility threshold. This paper aims to focus on outputting all the set of patterns having its utility value greater than or equal to the minimum utility threshold set by the user.

IV CHALLENGES IN HUIM

The following are the two main difficulties faced in HUIM

- The state space tree may contain very large number of itemset from which high utility itemset can be identified. For example, if a transaction database contains n distinct items, there will be 2n-1 possible itemset.

- Usually, High utility itemset are spread out in search space. In FIM this problem is solved using downward closure property, which will reduce the search space to find frequent itemset. But this property cannot be applied in finding HUI.

  To solve this problem a model was proposed, called Transaction-weighted utilization (TWU), which will improve the performance of the mining task. This TWU will support downward closure property in HUI technique.

V ALGORITHMS USED FOR HUIM

Various algorithms exist to extract high utility itemset. Some of the popular HUIM algorithms are UMining, IHUP, Two-phase, UP-Growth, HUP -Growth, MU-Growth, HUI-Miner, FHM, ULB-Miner, HUI Miner* and EFIM. All these algorithms take same input for processing and give same output after processing. They usually differ only in the way of processing the input to get the output. These algorithms differ on the following aspects [12].

- Whether they use BFS or DFS
- Whether internal or external type of representation used for the database.
- Method used to Explore next itemset from state space tree.
- Method used to calculate the utility of itemset to determine whether it satisfy the user specified minimum utility constraint or not.

The following factors have a great influence in determining the performance of the above high utility-based algorithms .

- Execution time
- Usage of memory
- Scalability factor
- How easily we can implement and extend the algorithm so that it can be used in other algorithms for data mining tasks.

VI CLASSIFICATION OF HUI ALGORITHMS

Algorithms used for mining HUI can be classified into two types. They are

- Two-Phase model algorithm
- One-Phase model algorithm

These two classes of algorithm differ in the number of candidates itemset generation [16].

A. Two-phase method:

There are many algorithms that work using two-phase method in which some are Two-Phase, IHUP, UP-Growth.

The basic idea behind two phase algorithm is that it can reduce the search space tree without missing high utility item sets. This can be achieved by applying an upper bound on utility measure. TWU (Transaction Weighted Utilization) measure is the proposed measure in Two-Phase method is used to prune the permutation tree. In this method, two phases are needed to perform high utility itemset mining. TWU for the itemset in search space are calculated during the first phase.

In second phase, we will traverse through the database and calculate the correct utility value of each candidate high utility itemset which were identified during first phase. if u(I)>=minutil, the itemset is considered as a high utility itemset.

Two-Phase method has both pros and cons. Main advantage of Two-phase method is that only low utility item will be eliminated from the search space. But the problem faced by the algorithm is that numerous candidates are generated which may cause memory and running time overhead [17] and minimum two database scanning is needed to generate HUI.

B. One Phase Algorithm:

One Phase Algorithm will overcome the drawbacks of the two-phase method by utilizing various data structures and various techniques to limit the number of candidates and to prune unpromising itemset. Here calculation of high utility itemset is done in a single phase. This model will output HUI while candidates are being generated. Which means that the item sets are immediately outputted as low utility itemset and high utility itemset, while creating the candidate itemset.

HUI-Miner [14,15], ULB-Miner, EFIM are some examples of one phase algorithms.

Algorithm	Search type	Nb of phases	DB representation	Extends
Two-Phase [59]	Breadth-first	Two	Horizontal	Apriori [2]
PB [47]	Breadth-first	Two	Horizontal	Apriori [2]
IHUP [5]	Depth-first	Two	Horizontal (prefix-tree)	FP-Growth [39]
UPGrowth(+) [79]	Depth-first	Two	Horizontal (prefix-tree)	FP-Growth [39]
HUP-Growth [52]	Depth-first	Two	Horizontal (prefix-tree)	FP-Growth [39]
MU-Growth [87]	Depth-first	Two	Horizontal (prefix-tree)	FP-Growth [39]
D2HUP [60]	Depth-first	One	Vertical (hyperstructure)	H-Mine [69]
HUI-Miner [58]	Depth-first	One	Vertical (utility-lists)	Eclat [91]
FHM [31]	Depth-first	One	Vertical (utility-lists)	Eclat [91]
mHUIMiner [70]	Depth-first	One	Vertical (utility-lists)	Eclat [91]
HUI-Miner* [71]	Depth-first	One	Vertical (utility-lists*)	Eclat [91]
ULB-Miner [17]	Depth-first	One	Vertical (buffered utility-lists)	Eclat [91]
EFIM [94]	Depth-first	One	Horizontal (with merging)	LCM [81]

Fig:1. Algorithms for HUIM [1]

In fig 1. A comparison of different HUIM algorithms is specified. Each algorithm compared based on the search types (BFS or DFS), data base representation (horizontal or vertical), number of phases (one or two phase), and same type of algorithms of frequent itemset mining.

VII VARIOUS EXTENSION TO HUIM

Researchers are concentrating to the following areas to find out solutions which can be used for various real-world applications.

A. Concise representation of High Utility Itemset

If the parameter for the algorithm called minimum utility threshold is set to a very low value, many patterns will be outputted to the user which will be time consuming and difficult for human to analyze. To overcome this drawback many researchers have designed algorithms for concise representation of HUI. These algorithms will make mining of HUI faster.

Algorithm	Patterns	Nb of phases	DB representation	Extends
MinFHM [29]	MinHUIs	One	Vertical (utility-lists)	FHM [31]
GHUI-Miner [30]	GHUIs	One	Vertical (utility-lists)	FHM [31]
CHUD [14]	CHUIs	Two	Vertical (utility-lists)	DCI_Closed [63]
CHUI-Miner [14]	CHUIs	One	Vertical (utility-lists)	DCI_Closed [63]
CLS-Miner [14]	CHUIs	One	Vertical (utility-lists)	FHM [31]
EFIM-Closed [94]	CHUIs	One	Horizontal (with merging)	EFIM [94]
GUIDE [75]	MHUIs	One	Stream	UPGrowth [79]
CHUI-Mine [89]	MHUIs	One	Vertical (utility-lists)	HUI-Miner [58, 71]

Fig:2. Algorithms for mining concise representation of HUI

Fig 2 represents some of the algorithms that efficiently discover concise representation of HUI and their characteristics.

B. Top-K High Utility Itemset Mining

User must specify the minutil threshold in traditional HUIM algorithms. The usage of memory, time needed, or the execution of algorithm is closely related to the user specified minutil. Top-K HUIM will help user to discover K -itemset that have highest utility in quantitative database [18].

C. High Utility Itemset Mining in Dynamic Database

Most of the traditional HUIM algorithms works on static database. They will not update the results when database is updated. Now various algorithms are developing for mining HUI from dynamic database. So, we will get updated result whenever any updates are done in database. Many algorithms have been developed to mine high utility itemset from infinite stream of transactions [19,20]

D. HUIM with Negative Utilities

Utility values in traditional HUIM are considered as positive. But in many real-world applications, the database may contain negative utility values. So efficient algorithms developing to handle this situation.

E. HUIM with Average Utility

HAUIM is considered as a variant to HUIM, which is used to find all the itemset in D. Here we use average-utility measure to discover the utility of itemset.

For an itemset I in a quantitative database D, we can calculate the average utility of I using the following equation. $au(I)=u(I)/|I|$.

The problem is to find all itemset in D having an average utility not less than user-specified minimum average utility threshold .

VIII CONCLUSION

HUIM is an active field of research since it has many real-world applications. Each one will introduce some effective structures to reduce unnecessary candidate generation and methods to speed up the HUI generation [17]. For many real- world applications, it needs to extend the basic concepts of HUIM which will create an extended version of HUIM. Now HUIM becomes the trending area of research. The Paper has alluded the algorithm HUIM along with some other algorithms which were used for mining HUI. An insight on the research conducted on these topics were also mentioned.

REFERENCES

[1] Shivam Agarwal “Data Mining: Fata Mining Concepts and Techniques” IEEE International Conference on Machine Intelligence Research and Advancement 2013.

[2] M. S. Chen, J. Han, and P. S. Yu, “Data mining: An overview from a database perspective,” IEEE Trans. Knowl. Data Eng.,vol. 8, no. 6, pp. 866–883, Dec. 1996.

[3] J. Han, J. Pei, Y. Yin, and R. Mao, “Mining frequent patterns without candidate generation: A frequent-pattern tree approach,” Data Mining Knowl. Discovery, vol. 8, no. 1, pp. 53–87, 2004.

[4] C. C. Aggarwal, Y. Li, J. Wang, and J. Wang, “Frequent pattern mining with uncertain data,” in Proc. 15th ACM SIGKDD Int.Conf. Knowl. Discovery Data Mining, 2009, pp. 29–38.

[5] R. Agrawal, T. Imieli_nski, and A. Swami, “Mining association rules between sets of items in large databases,” ACM SIGMOD Rec., vol. 22, no. 2, pp. 207–216, 1993.

[6] R. Agrawal, R. Srikant, et al., "Fast algorithms for mining association rules," in Proc. 20th Int. Conf. Very Large Data Bases, 1994, pp. 487–499.

[7] H. Mannila, H. Toivonen, and A. I. Verkamo, "Discovery of frequent episodes in event sequences," Data Mining Knowl. Discovery, vol. 1, no. 3, pp. 259–289, 1997.

[8] K. Y. Huang and C. H. Chang, "Efficient mining of frequent episodes from complex sequences," Inf. Syst., vol. 33, no. 1, pp. 96–114, 2008.

[9] A. Achar, S. Laxman, and P. Sastry, "A unified view of the prioribased algorithms for frequent episode discovery," Knowl. Inf.Syst., vol. 31, no. 2, pp. 223–250, 2012.

[10] A. Achar, A. Ibrahim, and P. Sastry, "Pattern-growth based frequent serial episode discovery," Data Knowl. Eng., vol. 87,pp. 91–108, 2013.

[11] P. Fournier-Viger, J. C. W. Lin, R. U. Kiran, Y. S. Koh, and R. Thomas, "A survey of sequential pattern mining," Data Sci. Pattern Recognit., vol. 1, no. 1, pp. 54 –77, 2017.

[12] P. Fournier-Viger ,J. Chun-Wei Lin,T. Truong-Chi,R. Nkambou," High-Utility Pattern Mining"Studies in Big Data 51, Springer 2019.

[13] Yao, H., Hamilton, H.J.:Mining itemset utilities from transaction databases. Data Knowl. Eng.59(3), 603–626 (2006).

[14] Liu, M., Qu, J.: Mining high utility itemsets without candidate generation. In: Proceedings of the 21st ACM International Conference Information and knowledge management, pp. 55–64.ACM (2012).

[15] Qu, J.-F., Liu,M., Fournier-Viger, P.: Efficient algorithms for high utility itemset mining without candidate generation. In: Fournier-Viger et al. (eds). High-Utility Pattern. Mining: Theory,Algorithms and Applications. Springer (2018).

[16] TOPIC: Top-k High-Utility Itemset Discovering Jiahui Chen, Member, IEEE, Shicheng Wan, Wensheng Gan, Member, IEEE,Guoting Chen, and Hamido Fujita, Senior Member, IEEE.

[17] Chongsheng Zhang,George Almpanidise,,Wanwan Wang, Changchang Liu,"An empirical evaluation of high utility itemset mining algorithms", Volume 101, Elsevier , 2018lnid.

[18] Fournier-Viger, Jerry Chun-Wei Lin, Tin Truong-Chi and Roger Nkambou P.A ,"Survey of High Utility Itemset Mining 2019.

[19] Duong, H., Ramampiaro, H., Norvag, K., Fournier-Viger, P., Dam, T.-L.: High utility drift detection in quantitative data streams. Knowl. Based Syst. 157(1), 34–51 (2018).

[20] Shie, B.-E., Yu, P.S., Tseng, V.S.: Efficient algorithms for mining maximal high utility itemset from data streams with different models. Expert Syst. Appl. 39(17), 12947–12960 (2012).