International Journal of Engineering Research and General Science Volume 4, Issue 1, January-February, 2016
ISSN 2091-2730

A Survey on Efficient Algorithms for Mining HUI and Closed Item sets

Mr. Mahendra M. Kapadnis[1], Mr. Prashant B. Koli[2]

1 PG Student, Kalyani Charitable Trust's Late G.N. Sapkal College of Engineering, Nashik, Maharashtra, India.

2 Assistant Professor, Kalyani Charitable Trust's Late G.N.Sapkal College of Engineering, Nashik, Maharashtra, India.

Abstract— High utility itemsets refer to the sets of items with high utility like profit in a database, and efficient mining of high utility itemsets plays a crucial role in many real-life applications and is an important research issue in data mining area. Mining high utility itemsets (HUIs) from large amount of databases is an essential data mining job, which refers to the finding of itemsets with high utilities or values (e.g. high profits). However, it may possible that exist too many HUIs to users, which also reduces the efficiency of the mining process. To accomplish high efficiency for the mining job and to provide a summarizing mining result to users from large amount of databases, we propose a new and innovative framework in this paper for mining closedþ high utility itemsets (CHUIs), which functions as a compact and lossless representation of HUIs. We propose three capable algorithms named AprioriCH (Apriori-based algorithm for mining High utility Closed þ itemsets), AprioriHC-D (AprioriHC algorithm with Discarding unpromising and isolated items) and CHUD (Closed þ High Utility Itemset Discovery) to find this representation. Later on, a scheme called DAHU (Derive All High Utility Itemsets) is proposed to recuperate all HUIs from the set of CHUIs without accessing the original database. Results which obtain from real and synthetic datasets indicates that the proposed algorithms are very efficient and that our methodologies reach a massive decrease in the number of HUIs. In addition, when all HUIs can be recuperated by DAHU, the combination of CHUD (Closed þ High Utility Itemset Discovery) and DAHU (Derive All High Utility Itemsets) overtakes the state-of-the-art algorithms for mining HUIs.

Keywords— Frequent itemset, high utility itemset, lossless and concise representation, closed itemset, utility mining, data mining.

INTRODUCTION

You can put the page in this format as it is and do not change any of this properties. You can copy and past here and format accordingly to the default front. It will be easy and time consuming for you.

IJERGS staff will revise and reformat if required

1. INTRODUCTION

In data mining Frequent itemset mining is a very important research issue. Market basket analysis is the mostly use one of its widely held applications, in which sets of items (itemsets) discovers that are frequently purchased collected by customers. The traditional model of Frequent itemset mining may finds a huge quantity of frequent itemsets with less profit and miss the data on valuable itemsets having less retailing frequencies. These problems are occurs due to Frequent itemset mining considers all items as having the same importance/unit profit/weight and it imagines that every item in a transaction looks in a binary form, i.e., an item doesn't indicate its purchase quantity in the transaction it may be in the transaction or may be not. Hence, Frequent itemset mining cannot fulfill the necessity of users who want to find itemsets with high values such as high profits.

Utility mining plays as an important role in data mining to solve the above issues. In utility mining, each item has a weight (e.g. unit profit) and may appear more than once in each operation (e.g. purchase quantity). The value of an itemset shows its importance, as it measured in terms of weight, profit, cost, quantity or other data depending on the users first choice. An itemset is named a high utility itemset (abbreviated as HUI) when its utility is no less than a user stated bottom utility threshold. A wide range of applications utility mining has such as website click stream analysis, cross-marketing analysis and biomedical areas.

2. BACKGROUND

Now a days High utility itemset (HUI) is providing a no- binary frequency values of items and different profit values for each item so it becomes a very important research theme in data mining. An incremental and interactive data mining deliver the capability to use earlier structures and mining results in order to less unwanted calculations when the database is modernized or the bottom value of threshold is change.

In Dec. 2009 C. F. Ahmed, S. K. Tanbeer, B.-S. Jeong, and Y.-K. Lee, proposes an efficient tree structures for high utility pattern mining in incremental databases [2] which has three novel tree structures to efficiently accomplish incremental and interactive High Utility Pattern (HUP) mining. The first tree structure, Incremental HUP Lexicographic Tree ($IHUP_L$-Tree), is prepared according to an item's lexicographic manner. It can capture the incremental data without any rearrangement process. The next one tree structure is the IHUP transaction frequency tree ($IHUP_{TF}$-Tree), which gains a compressed size by ordering items as per their transaction frequency (descending order). The third tree, IHUP-transaction-weighted utilization tree ($IHUP_{TWU}$-Tree) is build to decrease the mining period, grounded on the TWU cost of items in descending order. General performance studies illustrates that these tree structures are very efficient and scalable for incremental and interactive HUP mining

A huge collection of transactions having items, a basic mutual data mining difficulty is to extract the so-called frequent itemsets (i.e., sets of items presenting in at least a provided number of transactions). In 2003, J.-F. Boulicaut, A. Bykowski, and C. Rigotti, proposed a Free-sets which is a condensed representation of Boolean data for the approximation of frequency queries [3] in Data Mining Knowl. Discovery. The paper propose a structure called free-sets, from which we can estimate any itemset support (i.e., the number of transactions containing the itemset) and they validate this idea in the framework of adequate representations (H. Mannila and H. Toivonen, 1996. In Proc. of the Second International Conference on Knowledge Discovery and Data Mining (KDD'96), pp. 189–194). It displays that frequent free-sets can be efficiently take out using pruning strategies established for frequent itemset discovery, and that they can be used to estimate the backing of any frequent itemset. Testing on real dense data sets show a markable reduction of the size of the output at time of compared with standard frequent itemset extraction. Additionally, the experiments show that the extraction of frequent free-sets is still possible when the extraction of frequent itemsets becomes inflexible, and that the supports of the frequent free-sets can be used to approximate very closely the supports of the frequent itemsets. Finally, it shows the effect of this approximation on association rules (a popular kind of patterns that can be derived from frequent itemsets) and display that the corresponding errors persist very low in exercise.

Current studies on frequent itemset mining algorithms resulted in significant performance enhancements. However, if the minimal backing threshold is set too near to the ground, or the data is highly interrelated, the amount of frequent itemsets own can be prohibitively huge. To solve this type of problem, recently multiple type proposals have been made to built a summarizing demonstration of the frequent itemsets, as an alternative of mining all frequent itemsets. The main objective of this paper is to recognize redundancies in the set of all frequent itemsets and to deed these redundancies in order to decrease the result of a mining process. The paper display deduction rules to originate tight bounds on the provision of candidate itemsets. It also specify how the deduction rules permit for building a bottom level demonstration for all frequent itemsets. It also displays connections in this current proposal and recently available proposals for summarizing demonstrations and they provide the results of research on real-life datasets which show the usefulness of the deduction rules. In brief, the research even show that in multiple types of cases, first mining the summarized representation, and then making the frequent itemsets from this demonstration outperforms previously present frequent set mining algorithms.

In 2008, K. Chuang, J. Huang, and M. Chen proposed a Mining top-k frequent patterns in the presence of the memory constraint [5].

This paper explore a practicably remarkable mining job to retrieve *top-k* (*closed*) itemsets in the occurrence of the memory constraint. Precisely, as contrasting to most existing works that hardly focuses on improving the mining efficiency or on decreasing the memory size by best strength, It firstly try to mention the presented top memory size that can be used by mining frequent itemsets. To obey with the top bound of the memory intake, two efficient algorithms, called *MTK* and *MTK_Close*, are invented for mining frequent itemsets and *closed* itemsets, correspondingly, *without* mentioning the subtle bottom support. In its place, users only necessity is to give a more human-clear parameter, specifically the desired number of frequent (*closed*) itemsets *k*. In rehearsal, it is pretty challenging to constrain the memory consumption while also efficiently obtaining *top-k* itemsets. To effectively obtain this, *MTK* and *MTK_Close* are invented as level-wise finiding algorithms, where the number of candidates are to be generated-and-tested in every database scan will be limited. A unique search approach, named as δ-*stair search*, is used in *MTK* and *MTK_Close* to effectively allocate the available memory for testing candidate itemsets with different type of itemset-lengths, which leads to a small number of essential database scans. As demonstrated in the empirical observations on real data information and synthetic data information, instead of only providing the flexibility of striking a negotiation between the execution efficiency and the memory utilization, *MTK* and *MTK_Close* can both obtain top efficiency and have a constrained memory bound, giving the prominent advantage to be real-world algorithms of mining frequent patterns.

In 2003, R. Chan, Q. Yang, and Y. Shen, proposed mining high utility itemsets where mining high utility itemsets from a transactional database [6] refers to the finding of itemsets with high utility like profits. Even if a number of relevant algorithms have been proposed in current years, they incur the problem of creating a huge number of candidate itemsets for high utility itemsets. Such a huge number of candidate itemsets reduces the mining performance in terms of accomplishment time and space necessity. As the

situation may become worse when the database contains tons of long transactions or long high utility itemsets. This paper propose two algorithms, viz. utility pattern growth (UP-Growth) and UP-Growth+, for mining high utility itemsets with a set of effective policies for pruning candidate itemsets. The data of high utility itemsets is maintained in a tree-based data structure named utility pattern tree (UP-Tree) like that candidate itemsets may be generated efficiently with only two scans of database. The performance of UP-Growth and UP-Growth+ is equated with the state-of-the-art algorithms on multiple types of both real and synthetic data sets. Experimental outcomes show that the proposed algorithms, mainly UP-Growth+, not only decrease the number of candidates effectively but also outperform many types of algorithms substantially in terms of runtime, especially when databases contain multiple long transactions.

Data Mining can be well-defined as an activity that extracts some new nontrivial data contained in huge databases. Traditional data mining methods have concentrated mostly on detecting the statistical correlations among the items that are more frequent in the transaction databases. Also named as frequent itemset mining, these techniques were built on the rationale that itemsets which appear more frequently must be of more importance to the user in view of the business perspective .In this paper they focuses on an emerging area called Utility Mining which not only considers the frequency of the itemsets but also considers the utility related with the itemsets. The term utility denotes to the importance or the usefulness of the appearance of the itemset in transactions quantified in profit, sales like terms or any other user preferences. In High Utility Itemset Mining the aim is to recognize itemsets that have utility values more than a given utility threshold. In this paper a literature review of the present state of research and the various algorithms for high utility itemset mining is present.

In 1994, R. Agrawal and R. Srikant, proposed Fast algorithms for mining association rules in which they consider the problem of determining association rules among items in a huge database of sales transactions. Paper provide two new algorithms for solving the problem that are basically different from the known algorithms. Experimental assessment shows that these algorithms outperform the known algorithms by factors ranging from three for minor problems to more than an order of magnitude for huge problems. It also display how the best features of the two proposed algorithms can be joined into a hybrid algorithm, called AprioriHybrid. Scale-up research show that AprioriHybrid scales linearly with the number of transactions. AprioriHybrid also has outstanding scale-up assets in view of transaction size and the number of items in the database

3. PROPOSED SYSTEM

The Existing system is Frequent itemset mining (FIM) is a basic research topic in data mining. The market basket analysis is one of its popular applications, which related to the discovery of sets of items (itemsets) that are frequently purchased jointly by customers. Where, in this application, the old regular model of FIM may discover a huge amount of frequent but less revenue itemsets and miss the data on valuable itemsets having less retailing frequencies. That type of problems are occur because of the facts that (1) FIM rates all items as having the same importance/unit profit/weight and (2) it consider that every item in a transaction present in a binary form, i.e., an item can be either present or absent in a transaction, which does not shows its purchase quantity in the transaction. Hence, FIM cannot fulfill the requirement of users who desire to find itemsets having high utilities such as high profits.

HUI mining is not an easy job as the downward closure property in FIM does not grasp in utility mining. In different way we can say that, the search space for mining HUIs cannot be directly reduced as it is done in FIM because a superset of a less utility itemset can be a high utility itemset. Were proposed for mining HUIs, but they also present a huge number of high utility itemsets to users. A very huge number of high utility itemsets makes it problematic for the users to understand the results. It may also root the algorithms in way to make it inefficient in terms of time and memory necessity, or may run it out of memory. It is broadly acknowledged that the more high utility itemsets the algorithms create, the more processing they consume. The performance of the mining job reducses significantly for less minimum utility thresholds or when dealing with condensed databases.

3.1 Block diagram of the proposed system

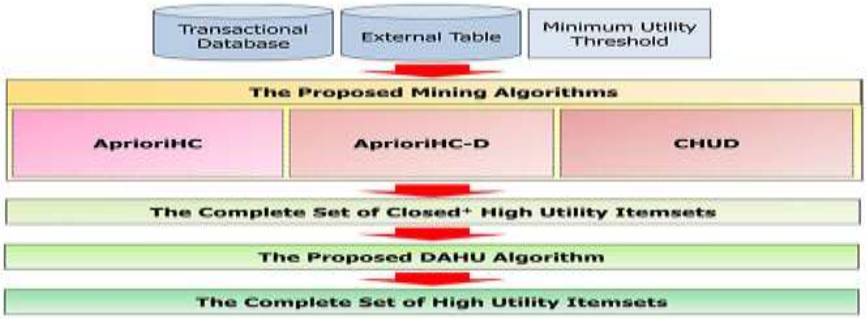


Fig. Block Diagram

4. MATHEMATICAL MODEL

4.1 Problem description

1) Input
2) Push Closed Property
3) Apriori HC Algorithm
4) Apriori HC D Algorithm
5) Recovery of HUI

Let the system be described by S,

S= {D, I, PCP, HC, HCD, R}

Where,

S: is a System.

D: is the set of Dataset.

I: Input.

PCP: Push Closed Property

HC: Apriori HC Algorithm

HCD: Apriori HC D Algorithm

R: Recovery of HUI

4.2 Activity

D= {d1, d2, … dn}

F= {f1, f2, … fn}

Y= { I, PCP, HC, HCD, R }

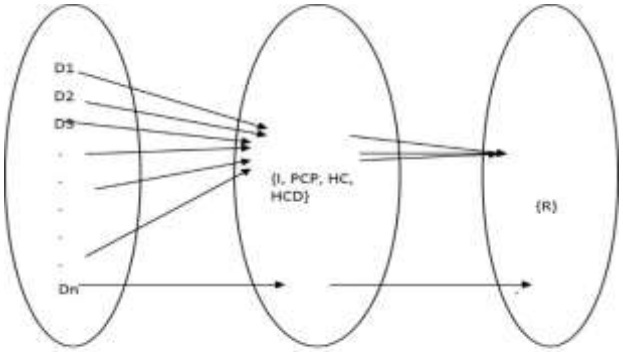
Where,

D is the Set of Dataset

F is the set of Function.

Y is a set of techniques use for Efficient Algorithms For Mining .

4.3 Vein Diagram



where,

D: is the set of Dataset.

I: Input.

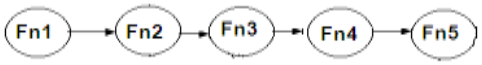
PCP: Push Closed Property.

HC: Apriori HC Algorithm.

HCD: Apriori HC D Algorithm.

R: Recovery of HUI.

4.4 State diagram



Where,

Fn1: Input

Fn2: Push Closed Property

Fn3: Apriori HC Algorithm

Fn4: Apriori HC D Algorithm

Fn5: Recovery

4.5 Functional Dependencies

	Fn1	Fn2	Fn3	Fn4	Fn5
Fn1	1	0	0	0	0
Fn2	0	1	0	0	0

Fn3	0	0	1	0	0
Fn4	0	0	0	1	0
Fn5	0	0	0	0	1

where,

Fn1: input

Fn2: push closed property

Fn3: apriori hc algorithm

Fn4: Apriori HC D Algorithm

Fn5: Recovery

5. CONCLUSIONS

In this paper, the difficulty of redundancy in high utility itemset mining by proposing a lossless and compact representation termed closedþ high utility itemsets is solved. To do the mining of this representation, three capable algorithms called AprioriHC (Apriori-based approach for mining High utility Closed itemset), AprioriHC-D (AprioriHC algorithm with Discarding unpromising and isolated items) and CHUID (Closedþ High Utility itemset Discovery). AprioriHC-D is an improved version of AprioriHC, which integrates approaches DGU and IIDS for pruning candidates. AprioriHC and AprioriHCD accomplish a breadth-first search for mining closedþ high utility itemsets from horizontal database, where else CHUD performs a depth-first search for mining closedþ high utility itemsets from vertical database. The strategies integrated in CHUD are efficient and unique and innovative. They have never been used for vertical mining of high utility itemsets and closed high utility itemsets. To efficiently recover all high utility itemsets from closedþ high utility itemsets, we proposed an efficient method termed DAHU (Derive All High Utility itemsets). The combination of CHUD and DAHU is also faster than UP-Growth when DAHU could be applied.

In the future, our aim is to combine many other compact representations like free sets, non-derivable frequent itemsets, Relative risk and odds ratio with high utility itemset mining.

REFERENCES:

[1] Vincent S. Tseng, Cheng-Wei Wu, Philippe Fournier-Viger, and Philip S. Yu, Fellow, "Efficient Algorithms for Mining the Concise and Lossless Representation of High Utility Itemsets", IEEE transactions on knowledge and data engineering, vol. 27, no. 3, March 2015.

[2] C. F. Ahmed, S. K. Tanbeer, B.-S. Jeong, and Y.-K. 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] J.-F. Boulicaut, A. Bykowski, and C. Rigotti, "Free-sets: A condensed representation Of Boolean data for the approximation of frequency queries," Data Mining Knowl. Discovery, vol. 7, no. 1, pp. 5–22, 2003.

[4] T. Calders and B. Goethals, "Mining all non-derivable frequent itemsets," in Proc. Int. Conf. Eur. Conf. Principles Data Mining Knowl. Discovery, 2002, pp. 74–85.

[5] 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.

[6] R. Chan, Q. Yang, and Y. Shen, "Mining high utility itemsets," in Proc. IEEE Int. Conf. Data Min., 2003, pp. 19–26.

[7] A. Erwin, R. P. Gopalan, and N. R. Achuthan, "Efficient mining of high utility itemsets from large datasets," in Proc. Int. Conf. Pacific- Asia Conf. Knowl. Discovery Data Mining, 2008, pp. 554–561.

[8] K. Gouda and M. J. Zaki, "Efficiently mining maximal frequent itemsets," in Proc. IEEE Int. Conf. Data Mining, 2001, pp. 163–170.

[9] T. Hamrouni, "Key roles of closed sets and minimal generators in concise representations of frequent patterns," Intell. Data Anal., vol. 16, no. 4, pp. 581–631, 2012.

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

[11] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," in Proc. 0th Int. Conf. Very Large Data Bases, 1994, pp. 487–499.