MINING TOP-K HIGH UTILITY ITEMSETS AND FREQUENT PATTERNS IN ONE PHASE

Shilna S
Department of Computer Science And Engineering Malabar Institute of Technology , Anjarakandy Kannur , Kerala - India
Email: Shilna94@gmail.com

Navya E K
Department of Computer Science And Engineering Malabar Institute of Technology , Anjarakandy Kannur, Kerala - India
Email:Navya.06rimaan@gmail.com

ABSTRACT

High utility itemsets (HUIs) mining is an trending topic in data mining,which refers to discovering all itemsets having a utility meeting a user-specified minimum utility threshold min_util.HUI aimed at finding itemsets that contribute most to the total utility.However, setting min_util appropriately is a difficult problem for users. Although a number of relevant algorithms have been proposed in recent years,but the problem is they produce large number of candidate itemsets for high utility itemsets.Generally utility mining adopt a two-phase, candidate generation approach,in the first phase,candidates of high utility patterns in the found out and in second phase, scan raw data one more time to identify high utility patterns from the candidates. The problem is that the number of candidates can be huge, which is the scalability and efficiency bottleneck.To solve this problem ,finds high utility patterns in a single phase without generating candidates.In this paper,address the above issues by proposing a new framework for top-k high utility itemset mining.TKO (mining Top-K utility itemsets in One phase) are proposed for mining itemsets without setting the min_util. Along with this frequent pattern is also generated using CR-Tree. CR Tree data structure is generated and then by using FP growth algorithm top-k frequent patterns is mined.Then the customer segmentation is done.Cluster the product based on customer.Then CBA (Customer Behaviour Analysis)is done and also emerging trends is given to the customers.By enabling companies to target specic groups of customers,find What are the top-k sets of products (i.e., itemsets) that contribute the highest profits to the company and according to the frequent pattern mining result, the merchandise in the supermarket is arranged together in the same place well-suited for customer.

Index Terms- *Utility mining, high utility itemset, frequent itemset mining, top-k pattern mining, Customer Behaviour Analysis.*

1 INTRODUCTION

Data mining is a powerful technology with high potential to help companies focus on the most important information in the collected data. It finds information within the data that queries and reports can’t effectively reveal. By performing data mining, knowledge,patterns,regularities or high-level information can be extracted from databases and viewed from different angles. This knowledge can be applied to decision making, process control,
information management and query processing Mining top k high utility itemsets from transactional databases is an important data mining task, which refers to the find itemsets with high utilities (e.g. high profits).The utility of an itemset represents its importance, which can be measured in terms of weight, value, quantity or other information depending on the user specification. An itemset is said to be high utility itemset (HUI) if its utility is no less than a user-specified minimum utility threshold min-util. HUI mining is essential to many applications such as streaming analysis, market analysis, mobile computing and biomedicine. Finding interesting patterns has been an important data mining task, and has a variety of applications, for example, genome analysis, condition monitoring, cross marketing, and inventory prediction, where interestingness measures play an important role. With frequent pattern mining , a pattern is regarded as interesting if its occurrence frequency exceeds a specified threshold. For example, mining frequent patterns from a shopping transaction database refers to finding of sets of products that are frequently purchased together by customers. However, a user’s interest may relate to many factors must not be expressed in terms of the occurrence frequency. For example, a supermarket manager may be interested in finding combinations of products with high profits, which relates to the unit profits and purchased quantities of products that are not

considered in frequent pattern mining.

2 BACKGROUND AND RELATEDWORK

Generally utility mining adopt a two-phase, candidate generation approach, that is,first find candidates of high utility patterns in the first phase, and then scan the raw data one more time to identify high utility patterns from the candidates in the second phase.Here d2HUP[1] algorithm is used. D2HUP, namely Direct Discovery of High Utility Patterns, which is an combination of the depth-first search of the reverse set enumeration tree, the pruning techniques.D2HUP does not perform in all applications, they are not developed for top-k high utility itemset mining and still suffer from the subtle problem of setting appropriate thresholds. Finding an appropriate minimum utility threshold by trial and error is a difficult process for users.Generally we have many algorithm,but many are two phase.So need to scan the database twice.It is time consuming and requires more memory usage.These algorithm does not give top k high utility itemsets also.They incur the problem of producing a huge number of candidate itemsets[2] for high utility itemsets. Such a large number of candidate itemsets decrease the mining performance[3] in terms of execution time and space requirement

Charu C. Aggarwal et.al provide a detailed survey of frequent pattern mining algorithms[4]. A wide variety algorithms will be used starting from Apriori. Many algorithms such as Eclat, TreeProjection, and FP-growth. In data mining, frequent pattern mining (FPM) is one of the most investigated problems in terms of computational and algorithmic development. Many algorithms have been proposed in last two decades, to solve frequent pattern mining or some of its variants, and the interest in this problem still persists .Different frameworks have been defined for mining frequent patterns. The common one is the support-based framework, in which itemsets with frequency above a given threshold are found. However, such itemsets may not usually represent interesting positive correlations between items. Consequently, alternative measures for interestingness have been defined in the literature . One of the main reasons for the high interest in mining frequent pattern algorithms is due to the computational challenge of the task. Even for moderate sized dataset, the search space of FPM is inordinate, which is exponential to the length of ttransactions in the dataset. This naturally cause challenges for itemset generation, when the support levels are low.It is critical to perform the analysis in a space- and time-efficient way. The main focus of work was to find FPM algorithms with better computational efficiency.In fact, the execution tree of all algorithms is different in terms of order in which the patterns are explored, and also check whether the counting work done for different candidates is independent of one another.

Market basket analysis[5] gives us insight into the merchandise by telling us which products tend to be purchased together and which are most enable to purchase .The market basket analysis is a powerful tool especially in retailing it is essential to discover large baskets, since it deals with thousands of items. FP-growth algorithm is an efficient algorithm for mining frequent patterns.

3 PROPOSED SYSTEM

In the proposed system Top-K High Utility Itemsets is mined by using a novel algorithm named TKO in one phase.In TKO min util is not set.Instead a border minimum utility threshold is used. TKO can raise the min_util threshold as quickly as possible, and further reduce as much as possible the number of candidates and intermediate low utility itemsets produced in the mining process.A new framework for top-k high utility itemset mining, where k is the desired number of HUIs. TKO (mining Top-K utility itemsets in One phase) are proposed for mining the complete set of top-k HUIs in databases without any need to specify the min_util threshold. the TKO algorithm uses a list-based structure named utility-list to store the utility information of itemsets in the database. TKO uses vertical data representation techniques to discover top-k HUIs in only one phase.It utilizes the search procedure of HUI-Miner and its utility-list structure. When an itemset is generated by TKO, its utility is calculated by its utility-list without scanning the original database. We first describe a basic version of TKO named TKOBase and then the advanced version, which includes several strategies to increase its efficiency.Many different types of data structure and algorithm have been proposed to extract frequent pattern from a large given database. CR algorithm is one of the fastest frequent pattern mining algorithm,Which can efficiently represent whole data structure over single scan of the database. We have proposed an efficient tree based structure CR Tree in terms of execution time and memory usage.Corelation Tree or CR Tree algorithms.. Even a huge database can be processed by CR Tree if out-of-date transactions are removed concurrently**.** CR Tree is better than FP tree.

```
Algorithm  - TKO
• Input:
1. u(P): utility-list for a prefix P
2. Class[P]: a set of itemsets w.r.t. the prefix P
3. ULS[P]: a set of utility-lists w.r.t.the prefix P
4. δ: border minimum utility threshold minutilBorder
• Output:
 1.TopK CI List: a list for storing candidate itemsets;

1: For each X = {x1,x2,,xL}∈Class[P] do
2: {If (SUM(X.iutils) >= δ)
3: { Raise min utilBorder by the strategy RUC
4: δ ← RUC(X,TopK − CI − List) }
5: If(SUM(X.iutils) + SUM(X.rutils) >= δ)
6: Class[X]← φ; ULS[X] ← φ
7: For each Y = {y1, y2,.. yL }∈Class[P]—yL > xLdo
8: Z ←XY
9: ul(Z)← Construct(ul(P), X, Y, ULS[P])
10: Class[X] ← Class[X] ∪ Z
11: ULS[X]← ULS[X] ∪ ul(Z)}
12: TopK-HUI-Search(X, ULS[X], Class[X],δ,TopK-CI-List)}}
```

The TKOBase algorithm takes as input the parameter k and a transactional database D in horizontal format. But if a database has already been transformed into vertical format such as initial utility-lists, TKOBase can directly use it for mining top-k HUIs.

TKOBase initially sets the min_utilBorder threshold to 0 and initializes a min-heap structure TopK-CI-List for maintain- ing the current top-k HUIs during the search. The algorithm then scans D twice to build the initial utility-lists ULs. Then, TKOBase explores the search space of top-k HUI using a procedure that we name Top K-HUI-Search. It is the combination of a novel strategy named RUC (Raising threshold by Utility of Candidates) with the HUI-Miner search procedure [14]. During the search, TKOBase updates the list of current top-k HUIs in TopK-CI-List and gradual- ly raises the min_utilBorder threshold by the information of TopK-CI-List. When the algorithm terminates, the TopK- CI-List captures the complete set of top-k HUIs in the database.
For each L-itemset X = {x1, x2,…, xL} generated by the search procedure, if its utility is no less than min_utilBorder, the proposed RUC strategy is applied to raise min_utilBorder, RUC is performed as follows. First, X is added into TopK- CI-List. Then, if EU(X) is no less than min_utilBorder and there are more than k itemsets already in TopK-CI-List, min_utilBorder is raised to the utility of the k-th itemset in TopK-CI-List. The remaining itemsets having a utility low- er than min_utilBorder are removed. This ensures that all and only the top-k HUIs are kept. After the above process, the strategy raises the border minimum utility threshold. Fig. 5 shows the pseudo code of TopK-HUI-Search pro- cedure. It continues mining itemsets that are concatena- tions of an itemset X if the sum of iutils and rutils of X is no less than min_utilBorder (Line 6). Two ordered sets Class[X] and ULS[X] are created to respectively store the concatenations of X and their utility-lists (Line 7). For each itemset Y = {y1, y2,…, yL} in Class[P] (yL ≻ xL and P = {y1, y2,…, , yL-1 }), we create a candidate itemset Z = X□Y by concatenating X with yL and uses the Construct proce- dure to construct utility-list of Z (i.e., ul(Z)) (Line 9-10). Then, Z and ul(Z) are respectively added to Class[X] and ULS[X] (Line 11-12). After processing each itemset in Class[P], the procedure TopK-HUI-Search is called with X, Class[X], min_utilBorder and TopK-CI-List to consider (L+1)- itemsets that are concatenations of X. This recursive process continues until no candidate itemset is found.

RUC: Raising the threshold by the Utilities of Candidates). This strategy can be incorporated with any one-phase mining algorithm where itemsets are found with their utilities. It adopts the TopK-CI-List structure to main- tain top-k HUIs, where itemsets are sorted by descending order of utility. Initially, TopK-CI-List is empty. When an itemset X is found by the search procedure and its utility is no less than min_utilBorder, X is added to TopK-CI-List. If there are more than k itemsets already in TopK-CI-List, min_utilBorder can be safely raised to the utility of the k-th itemset in TopK-CI-List. After that, itemsets having a utility lower than the raised min_utilBorder are removed from TopK- CI-List.

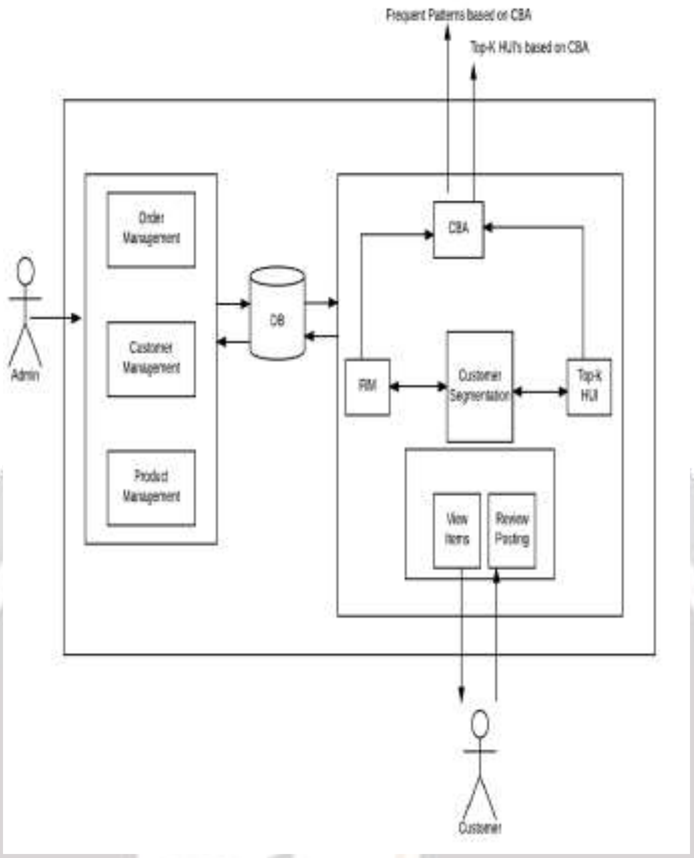

FIGURE 1 System Architecture

4 CONCLUSION

This work presents a novel algorithm named TKO of mining for high utility itemsets.Generally utility mining adopt a two-phase, candidate generation approach, that is, in the first phase,candidates of high utility patterns in the found out and in second phase, scan raw data one more time to identify high utility patterns from the candidates..It can discover top-k HUIs in only one phase. It utilizes the search procedure of HUI Miner and its utility-list structure. Whenever an itemset is generated by TKO, its utility is calculated by its utility-list without scanning the original database.Along with this frequent patterns is also mined using a efficient algorithm CR Tree.CR Tree extends the idea of FP- Tree to improve storage compression and allow frequent pattern mining without generation of candidate itemsets. In CR Tree both utility and support count is checked. CR Tree data structure is generated and then by using FP growth algorithm top-k frequent patterns is mined.Then the customer segmentation is done.By enabling companies to targe tspecic groups of customers, acustomersegmentation model allows for the eective allocation of marketing resources and the maximization of cross and up-selling opportunities.Cluster the product based on customer.Then CBA (Customer Behaviour Analysis)is done and also emerging trends is given to the customers.

.

REFERENCES

[1] Junqiang Liu, Ke Wang, and Benjamin CM Fung. Mining high utility patterns in one phase without generating candidates. IEEE Transactions on Knowledge and Data Engineering, 28(5):1245–1257, 2016.
[2] GuangzhuYu, KeqingLi, andShihuangShao. Mininghighutilityitem-sets in large high dimensional data. In Proceedings of the 1st internationalconference on Forensic applications and techniques in telecommunications,information, and multimedia and workshop, page 47. ICST (Institute foComputer Sciences, Social-Informatics and Telecommunications Engineering), 2008..
[3] Vincent S Tseng, Bai-En Shie, Cheng-Wei Wu, and S Yu Philip.Efficient algorithms for mining high utility itemsets from transactional databases. IEEE transactions on knowledge and data engineering,2013.
[4]JiaweiHan,JianPei,YiwenYin,andRunyingMao.Miningfrequentpat-terns without candidate generation: A frequent-pattern tree approach.Data mining and knowledge discovery, 8(1):53–87, 2004.
[5] Liu Yongmei and Guan Yong. Application in market basket researchbased on fp-growth algorithm. In Computer Science and InformationEngineering, 2009 WRI World Congress on, volume 4, pages 112–115, 2009.