Mining High Utility Patterns in One Phase for Big Data

Priti Deshmukh*, Prof. A. S. More

Miss.Priti H Deshmukh, Computer,
JSPM NTC, PUNE, INDIA-411041
pritideshmu.8991@gmail.com[1]

[2]Assistant professor A. S. More, Computer,
JSPM NTC, PUNE, INDIA-411041
anjalimore25@gmail.com

---***---

Abstract – Utility mining is a new emerging technology of data mining, but Utility mining does not consider the interestingness measure. High utility pattern growth approach is a look ahead strategy, and a linear data structure. Here linear data structure enables computing a tight bound for powerful pruning search space and to directly identify high utility patterns in an efficient and scalable way. In this it targets the root cause with prior algorithms. Now days, high utility pattern (HUP) mining is one of the most important research issues in data mining due to its ability to consider the no binary frequency values of items in transactions and different profit values for every item. But, incremental and interactive data mining provide the ability to use previous data structures and mining results in order to reduce unnecessary calculations when a database is updated, or when the minimum threshold is changed. In this analytical study, novel tree structures are proposed to efficiently perform incremental and interactive HUP mining.

Keywords—data mining, utility mining, high utility patterns, frequent patterns, pattern mining

1. INTRODUCTION

Frequent pattern mining is provided with the solution, candidate set generation and test paradigm of prior. It has many drawbacks like it requires multiple database scans and generates many candidate item sets. This problem is solved by growth approach by introducing a prefix-tree (FP-tree)-based algorithm without candidate set generation and testing. As frequent pattern mining plays an important role in data mining applications, its two limitations are given as, first, it treats all items with the same importance/weight/price and, second one is, in one transaction, each item appears in a binary (0/1) form, i.e., either it is present or absent. Since, in the real world, each item in the supermarket has a different importance/price and one customer can buy multiple copies of an item. So, items having high and low selling frequencies may have low and high profit values, respectively. Take a example as, some frequently sold items such as bread, milk, and pen may have lower profit values compared to that of infrequently sold higher profit value items such as gold ring and gold necklace. Therefore, finding only traditional frequent patterns in a database cannot fulfill the requirement of finding the most valuable item sets/customers that contribut1e to the major part of the total profits in a retail business. This gives the motivation to develop a mining model to discover the item sets/customers contributing to the majority of the profit. Now days, a utility mining model was defined to discover more important knowledge from a database. Here the importance of an item set by the concept of utility is measured. The dataset with no binary frequency values of each item in transactions, and also with different profit values of each item is handled. Therefore, utility mining represents real world market data. According to utility mining, several important business area decisions like maximizing the revenue or minimizing the marketing or inventory costs can be considered and knowledge about item sets/customers contributing to the majority of the profit can be discovered. But in real world retail market, takes the biological gene database and web click streams, also there is different importance of each gene or web site and their occurrences are not limited to a 0/1 value. Other application areas, such as stock tickers, network traffic measurements, web server logs, data feeds from sensor networks, and telecom call records can have similar solutions. It is not suitable for large databases e.g. big data. Earlier studies deals with only small datasets, here big data is considered where data is having properties like variety , variability, veracity, volume, velocity, etc. Utility mining is done for big data which improves the efficiency.

2. RELATED WORK

High utility pattern mining problem is related to frequent pattern mining. Here, we will study how prior works for frequent pattern mining and see how it relates to our work.

Frequent Pattern Mining

Frequent pattern mining discovers all patterns whose supports are no less than a user defined minimum support threshold. Frequent pattern mining holds the anti-monotonicity property i.e., the support of a superset of a pattern is no more than the support of the pattern. Algorithms for mining frequent patterns as well as algorithms for mining high utility patterns are breadth-first search, depthfirst search, and hybrid search.

This paper uses a depth-first strategy because breadth-first search is typically more memory intensive and more likely to exhaust main memory and thus it is slower. Also, algorithm depth-first searches a reverse set enumeration tree, which can be thought of as exploring a right-to-left in a reverse lexicographic order.

3. PROBLEM STATEMENT

Prior algorithm works for this problem i.e., with a two-phase candidate generation approach. But it is having one exception that is inefficient and not scalable with large databases. The two-phase approach has scalability issue due to the huge number of candidates. This analytical study proposes a new algorithm, d2HUP, for utility mining with the item set share framework, which finds high utility patterns in big data without candidate generation.

4. MOTIVATION

From the problems of utility mining, utility mining with the item set share framework is a hard one as it does not consider interestingness measure. Prior works for this problem with a two-phase candidate generation approach. But it is having one exception that is inefficient and not scalable with large databases. The two-phase approach suffers from scalability issue due to the huge number of candidates. To implement an effective share Framework of using new algorithm, d2HUP, for utility mining with the item set share framework, which finds high utility patterns in big data without candidate generation.

5. EXISTING SYSTEM

Here the algorithm, d2HUP, i.e. Direct Discovery of High Utility Patterns, which is an integration of the depth-first search of the reverse set enumeration tree, which prune the techniques that drastically reduces the number of patterns to be enumerated, and a novel data structure that enables efficient computation of utilities and upper bounds.

```
Algorithm 1: d2HUP($D,XUT, minU$)
build $TS(\{\})$ and $\Omega$ from $D$ and $XUT$
$N \leftarrow$ root of reverse set enumeration tree
DFS($N, TS(pat(N)), minU, \Omega$)
Subroutine: DFS($N, TS(pat(N)), minU, \Omega$)
if $u(pat(N)) \geq minU$then output $pat(N)$
$W \leftarrow \{i|i \prec pat(N) \wedge uBitem(i, pat(N)) \geq minU\}$
if $Closure(pat(N),W, minU)$ is satisfied
    then output nonempty subsets of $W \cup pat(N)$
else if $Singleton(pat(N),W, minU)$ is satisfied
    then output $W \cup pat(N)$ as an HUP
else foreachitem $i \in W$ in $\Omega$ do
    if $uBfpe(\{i\} \cup pat(N)) \geq minU$
        then $C \leftarrow$ the child node of $N$ for $i$
        $TS(pat(C)) \leftarrow$ Project($TS(pat(N)), i$)
        DFS($C, TS(pat(C)), minU, \Omega$)
end foreach
```

6.PROPOSED SYSTEM:

We can overcome the disadvantage of the existing method. In the existing system a single dataset is used for utility mining. But this is not suitable for large databases. So here we will overcome this problem by taking big data as a input. So for this first of all we need to do the parallel mining. So we need to partition the whole mining tasks into smaller independent subtasks and mining them independently and finally combining the results. So for the high utility mining we will provide a big data as a input for d2HUP algorithm.

CONCLUSION

The system is designed to implement new d2HUP algorithm for utility mining with the item set share framework, which finds high utility patterns without candidate generation. This approach is used to enhance the significance by the look ahead strategy that identifies high utility patterns without enumeration.

REFERENCES

[1] C. H. Cai, A. W. C. Fu, C. H. Cheng, and W. W. Kwong, “Mining association rules with weighted items,” in InternationalDatabase Engineering and Applications Symposium. IEEE, 68-77 1998.

[2] R. Agrawal, T. Imielinski, and A. Swami, “Mining association rules between sets of items in large databases,” in SIGMOD. ACM, 1993, pp. 207–216.

[3] R. Agrawal and R. Srikant, “Fast algorithms for mining association rules,” in VLDB, 1994, pp. 487–499.

[4] 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 TKDE, vol. 21, no. 12, pp. 1708– 1721, 2009.

[5] R. Bayardo and R. Agrawal, “Mining the most interesting rules,” in SIGKDD. ACM, 1999, pp. 145–154.

[6] F. Bonchi, F. Giannotti, A. Mazzanti, and D. Pedreschi, “Exante: A preprocessing method for frequent-pattern mining,” IEEEIntelligent Systems, vol. 20, no. 3, pp. 25–31, 2005.

[7] F. Bonchi and B. Goethals, “Fp-bonsai: The art of growing and pruning small fp-trees,” in PAKDD, 2004, pp. 155–160.

[8]G. Zhou, F. Liu, Y. Liu, S. He, and J. Zhao, \Statistical machine translation improves question retrieval in community question answering via matrix factorization", in ACL, 2013, pp. 852-861.

[9] A. Singh, \Entity based q and a retrieval", in EMNLP, 2014, pp. 1266-1277.