IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 4, April 2017

Mining High Utility Itemsets using Up-Tree Algorithm

S. Kalaiselvi, M.Sc. M.Phil., B.Ed.[1], S. Nithya Kalarani, M.Sc. M.Phil., B.Ed.,[2]

Assistant Professor, Department of CS & IT, Theni Kammavar Sangam College of Arts & Science, Theni[1]

Assistant Professor, Department of CS & IT, Theni Kammavar Sangam College of Arts & Science, Theni[2]

Abstract: Mining high utility itemset from XML database refers to the discovery of itemsets with high utility like profits. Although a number of relevant approaches have been proposed in recent years, they incur the problem of producing large number of candidate itemset for high utility itemsets. Such a large number of candidate itemset degrades the mining performance in terms of execution time and space requirement. The situation may become worse when the database contains large number of long transactions or long high utility itemsets (HUIs).The algorithm used here is UP-Growth (Utility Pattern Growth) for mining high utility itemsets with a set of techniques for pruning candidate itemsets. The information of high utility itemsets is maintained in a special data structure named UP-Tree (Utility Pattern Tree) such that the candidate itemsets can be generated efficiently with only two scans of the database. UP-Growth not only reduces the number of candidates effectively but also out performs other algorithms substantially in terms of execution time, especially when the database contains lots of long transactions. We can efficiently store and retrieve the data's in and from the XML databases than relational database.

Keywords: Itemset Mining, UP-Growth, LP-Tree.

1. INTRODUCTION

DATA mining is the process of revealing nontrivial, previously unknown and potentially useful information from large databases. Discovering useful patterns hidden in a database plays an essential role in several data mining tasks, such as frequent pattern mining, weighted frequent pattern mining, and high utility pattern mining. Among them, frequent pattern mining is a fundamental research topic that has been applied to different kinds of databases, such as transactional databases, streaming databases, and time series databases, and various application domains, such as bioinformatics, Web click-stream analysis, and mobile environments. Nevertheless, relative importance of each item is not considered in frequent pattern mining.

In this framework, weights of items, such as unit profits of items in transaction databases, are considered. With this concept, even if some items appear infrequently, they might still be found if they have high weights. However, in this framework, the quantities of items are not considered yet. Therefore, it cannot satisfy the requirements of users who are interested in discovering the itemsets with high sales profits, since the profits are composed of unit profits, i.e., weights, and purchased quantities. In view of this, utility mining emerges as an important topic in data mining field. Mining high utility itemsets from databases refers to finding the itemsets with high profits. Here, the meaning of itemset utility is interestingness, importance, or profitability of an item to users. Utility of items in a transaction database consists of two aspects:

1) The importance of distinct items, which is called external utility,

2) The importance of items in transactions, which is called internal utility. Utility of an itemset is defined as the product of its external utility and its internal utility. An itemset is called a high utility itemset if its utility is no less than a user-specified minimum utility threshold; otherwise, it is called a low-utility itemset. Mining high utility itemsets from databases is an important task has a wide range of applications such as website click stream analysis, business promotion in chain hypermarkets, cross marketing in retail stores, online e-commerce management, mobile commerce environment planning, and even finding important patterns in biomedical applications.

However, mining high utility itemsets from databases is not an easy task since downward closure property in frequent itemset mining does not hold. In other words, pruning search space for high utility itemset mining is difficult because a superset of a low-utility itemset may be a high utility itemset. A naive method to address this problem is to enumerate all itemsets from databases by the principle of exhaustion. Obviously, this method suffers from the problems of a large search space, especially

When databases contain lots of long transactions or a low minimum utility threshold is set. Hence, how to effectively prune the search space and efficiently capture all high utility itemsets with no miss is a crucial challenge in utility mining.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 4, April 2017

A data structure named UP-Tree was proposed for maintaining the information of high utility itemsets. PHUIs can be efficiently generated from UP-Tree with only two database scans. Moreover, we developed several strategies to decrease overestimated utility and enhance the performance of utility mining. In the experiments, both real and synthetic data sets were used to perform a thorough performance evaluation. Results show that the strategies considerably improved performance by reducing both the search space and the number of candidates.

2. RELATED WORKS

Potential high utility itemsets (PHUIs) are found first, and then an additional database scan is performed for identifying their utilities. However, existing methods often generate a huge set of PHUIs and their mining performance is degraded consequently. This situation may become worse when databases contain many long transactions or low thresholds are set. The huge number of PHUIs forms a challenging problem to the mining performance since the more PHUIs the algorithm generates, the higher processing time it consumes
Fast Algorithms for Mining Association Rules [1] We consider the problem of discovering association rules between items in a large database of sales transactions. We present two new algorithms for solving this problem that are fundamentally di erent from the known algorithms. Experiments with synthetic as well as real-life data show that these algorithms outperform the known algorithms by factors ranging from three for small problems to more than an order of magnitude for large problems. Mining Sequential Patterns [2] We introduce the problem of mining sequential patterns over such databases. We present three algorithms to solve this problem, and empirically evaluate their performance using synthetic data. Two of the proposed algorithms, AprioriSome and AprioriAll, have comparable performance, albeit AprioriSome performs a little better when the minimum number of customers that must support a sequential pattern is low. Scale-up experiments show that both AprioriSome and AprioriAll scale linearly with the number of customer transactions. Efficient Tree Structures for High Utility Pattern Mining in Incremental Databases [3] high utility pattern (HUP) mining is one of the most important research issues in data mining due to its ability to consider the nonbinary frequency values of items in transactions and different profit values for every item. On the other hand, 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. Mining Association Rules with Weighted Items [4] we generalize this to the case where items are given weights to reflect their importance to the user. The weights may correspond to special promotions on some products, or the profitability of different items. We can mine the weighted association rules with weights. Mining High Utility Itemsets Traditional association rule mining algorithms [5] a novel idea of top-K objective-directed data mining, which focuses on mining the top-K high utility closed patterns that directly support a given business objective. To association mining, we add the concept of utility to capture highly desirable statistical patterns and present a level-wise item-set mining algorithm. Mining Weighted Sequential Patterns in a Sequence Database with a Time-Interval Weight [6] In general sequential pattern mining, the generation order of data elements is considered to find sequential patterns. However, their generation times and time-intervals are also important in real world application domains. Therefore, time-interval information of data elements can be helpful in finding more interesting sequential patterns. Efficient Data Mining for Path Traversal Patterns [7] a new data mining capability that involves mining path traversal patterns in a distributed information-providing environment where documents or objects are linked together to facilitate interactive access.

3. PROPOSED SYSTEM

In this project, we propose a novel tree structure to solve the limitation. Our new structure, LP-tree (Linear Prefix – Tree) is composed of array forms and minimizes pointers between nodes. In addition, LP-tree uses minimum information required in mining process and linearly accesses corresponding nodes.

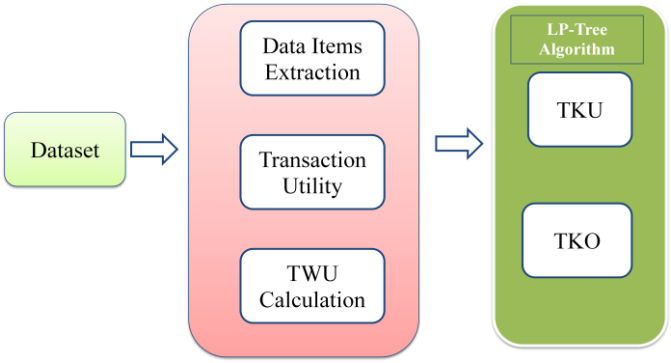


Fig :- Architecture Diagram

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 4, April 2017

We also suggest an algorithm applying LP-tree to the mining process. LP-tree has the following structure:
(1) Header list consisting of item-names, supports, and node links,
(2) Linear Prefix Node (LPN) for storing frequent items of each transaction and a corresponding header, and
(3) Branch Node List (BNL) including information of branch nodes and their child nodes.

3.1 Extracting data items

In the first module, the user can choose the dataset to get high utility itemsets. After choosing the dataset, the data items should be extracted from the transactions dataset. The data items are individual items which occurs in the transaction.

3.2 Calculating TU (Transaction Utility)

After constructing parse tree for the XML database. The Transaction Utility(TU) of each items is calculated. The transaction Utility is calculated by using the value of profit and its quantity.
Transaction Utility=Profit*quantity

3.3. Finding TWU (Transaction Weighted Utility)

After calculating Transaction Utility(TU) of each items from the parse tree. We need to find Transaction Weighted Utility(TWU) for each items from the calculated Transaction Utility(TU).The TWU of itemset whose value is less than the given threshold value are discarded by pruning items .The discarded itemsets are called unpromising itemset they don't yield more utility to the user. The itemset whose TWU value is less than minimum utility is called unpromising itemset , otherwise promising itemset.

3.4 TKO Tree Formation (LP Tree)

TKO (mining Top-k utility itemsets in One phase). It can discover top-k HUIs in only one phase. It utilizes the basic 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. TKO with LP Tree is formed with TWU, transactions and data items.

4. METHODOLOGY

UP-Growth algorithm

Input: UP-Tree Tx, Header Table HTx, minimum utility threshold t, Item set $I = \{i1, i2, \ldots, ik\}$. Process:

1. For each entry ik in HTx do
2. Trace links of each item. And calculate sum of node utility$nusum$.
3. If $nusum \geq$ t
4. Generate Potential High Utility Itemset (PHUI) $Y = X \cup ik$
5. Put Potential Utility of ik as approximated utility of Y
6. Construct Conditional Pattern Based HTY.
7. Put local promising items intoHTY.
8. Apply Discarding Local Unpromising (DLU) to minimize path utilities of paths.
9. Apply DLU with $Insert\_Recognized\_Path$ to insert path intoTY.
10. If $TY \neq \emptyset$ then call to UP-Growth.
11. End if
12. End for. Output: All PHUI's in T

In UP-Growth, minimum item utility table is used to reduce the overestimated utilities. In UP-Growth+ algorithm we replace Discarding Local Unpromising (DLU) with Discarding Node Utility (DNU), DLN is replace with Decreasing local Node utilities for the nodes of local UP-Tree (DNN) and Insert_Recognized_Path is replace by Insert_Recognized_Path, When a path is retrieved, minimal node utility of each node in the path is also retrieved in the data mining process . Thus, the minimum item utility can be simply replaced with minimal node utility.

5. EXPERIMENTAL RESULT

In order to verify the performance of the proposed algorithm, we compare it with existing algorithm. These algorithms are performed on a computer with a 2.00GHz processor and 512MB memory, running windows vista. The program is developed by Java with Mysql. We present experimental results using the database. The experimental result is showed in Figures. As shown in the Figure, proposed UP_growth with LP tree algorithm is more super than existing algorithm ,because it dosen't need to generate 2-candidate itemsets and reduce the search space, and proposed algorithm dosen't need to much extra spaces on the mining process, so proposed algorithm has a better space scalability
Compared to existing algorithms our performance is increased. The below tables represent the accurate values of current process and existing values.

IJARCCE

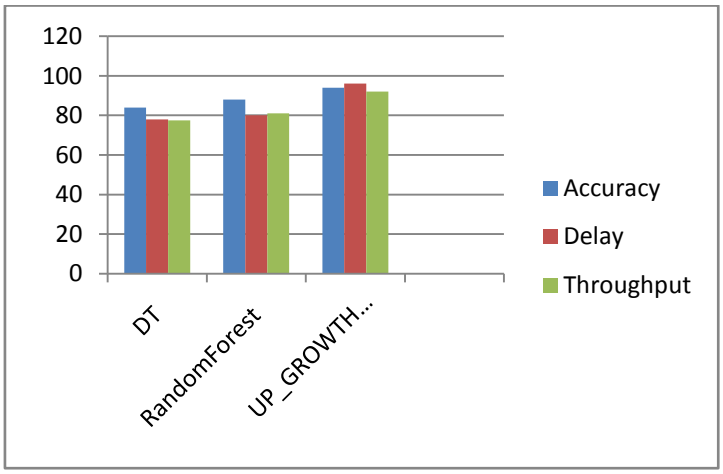
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 4, April 2017

Table 1 : Performance Table

Technique	Accuracy	Delay	Throughput
Decision Tree	84%	78%	77.5%
Random Forest	88%	80%	81%
UP Growth with LP tree	94%	96%	92%

The accuracy rate obtained by applying the classification algorithms on the data sets



The individual accuracy rates obtained from different feature selection methods on the classifier. Different feature selection metrics are applied on the classifier.

6. CONCLUSION

We have proposed two efficient algorithms named UP-Growth and UP-Growth+for mining high utility itemsets from transaction databases. A data structure named UP-Tree was proposed for maintaining the information of high utility itemsets. PHUIs can be efficiently generated from UP-Tree with only two database scans. Moreover, we developed several strategies to decrease overestimated utility and enhance the performance of utility mining. In the experiments, both real and synthetic data sets were used to perform a thorough performance evaluation. Results show that the strategies considerably improved performance by reducing both the search space and the number of candidates. Moreover, the proposed algorithms, especially UP-Growth+, outperform the stateof-the-art algorithms substantially especially when databases contain lots of long transactions or a low minimum utility threshold is used.

REFERENCES

[1] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules,"Proc. 20th Int'l Conf. Very Large Data Bases (VLDB),pp. 487-499, 1994.
[2] R. Agrawal and R. Srikant, "Mining Sequential Patterns,"Proc. 11th Int'l Conf. Data Eng.,pp. 3-14, Mar. 1995.
[3] 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. Knowledge and Data Eng., vol. 21, no. 12, pp. 1708-1721, Dec. 2009.
[4] C.H. Cai, A.W.C. Fu, C.H. Cheng, and W.W. Kwong, "Mining Association Rules with Weighted Items,"Proc. Int'l Database Eng. and Applications Symp. (IDEAS '98),pp. 68-77, 1998.
[5] R. Chan, Q. Yang, and Y. Shen, "Mining High Utility Itemsets," Proc. IEEE Third Int'l Conf. Data Mining,pp. 19-26, Nov. 2003.
[6] J.H. Chang, "Mining Weighted Sequential Patterns in a Sequence Database with a Time-Interval Weight,"Knowledge-Based Systems, vol. 24, no. 1, pp. 1-9, 2011.
[7] M.-S. Chen, J.-S. Park, and P.S. Yu, "Efficient Data Mining for Path Traversal Patterns,"IEEE Trans. Knowledge and Data Eng.,vol. 10, no. 2, pp. 209-221, Mar. 1998.
[8] C. Creighton and S. Hanash, "Mining Gene Expression Databases for Association Rules," Bioinformatics,vol. 19, no. 1, pp. 79-86, 2003.
[9] M.Y. Eltabakh, M. Ouzzani, M.A. Khalil, W.G. Aref, and A.K. Elmagarmid, "Incremental Mining for Frequent Patterns in Evolving Time Series Databases," Technical Report CSD TR#08-02, Purdue Univ., 2008.
[10] A. Erwin, R.P. Gopalan, and N.R. Achuthan, "Efficient Mining of High Utility Itemsets from Large Data Sets,"Proc. 12th Pacific-Asia Conf. Advances in Knowledge Discovery and Data Mining (PAKDD), pp. 554-561, 2008.
[11] E. Georgii, L. Richter, U. Ru¨ckert, and S. Kramer, "Analyzing Microarray Data Using Quantitative Association Rules,"Bioinformatics,vol. 21, pp. 123-129, 2005.
[12] J. Han, G. Dong, and Y. Yin, "Efficient Mining of Partial Periodic Patterns in Time Series Database,"Proc. Int'l Conf. on Data Eng., pp. 106-115, 1999.