Volume 2, No. 5, Sept-Oct 2011

ISSN No. 0976-5697

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Mining Regular Patterns in Transactional Databases using Vertical Format

G. Vijay Kumar*
School of Computing
K L University
Andhra Pradesh, India
gvijay_73@yahoo.co.in

M. Sreedevi
School of Computing
K L University
Andhra Pradesh, India
msreedevi_27@yahoo.co.in

NVS. Pavan Kumar
School of Computing
K L University
Andhra Pradesh, India
nvspavankumar@gmail.com

Abstract: Discovering interesting patterns in Transactional databases is a challenging area in data mining and knowledge discovery research. Recently, temporal regularity of a pattern was treated as an important criterion in several online applications like market basket analysis, network monitoring, gene data analysis, web page sequence and stock market. Although there have been some efforts done in finding *regular* patterns in a transactional database, no such method has been proposed yet by using vertical data format with one database scan. Therefore, in this paper we develop a new method using VDRP-table to generate the complete set of *regular* patterns in a transactional database for a user given regularity threshold. Our experiment results show that this method is efficient in both time and memory to find *regular* patterns.

Keywords: Data mining, Vertical databases, Regular patterns, Transactional database.

I. INTRODUCTION

Frequent Pattern Mining [1, 2] is one of the fundamental and essential area in Data Mining research. It finds patterns that appear frequently in a database. Several algorithms have been proposed so far to mine frequent patterns in a transactional database. However, the significance of a pattern may not always depend upon the occurrence frequency of a pattern (i.e., support). The significance of a pattern may also depend upon other occurrence characteristics such as temporal regularity of a pattern. For example, in a retail market some products may have regular demand than other products. To know how regularly a product has been sold is essential rather than the frequency of a product. So finding patterns at regular intervals also plays an important role in data mining.

Apriori algorithm [3] is a classical algorithm proposed by R. Agarwal and R. Srikanth in 1993 for mining frequent item sets for Boolean association rules. The algorithm uses prior knowledge and employs an iterative approach known as a level –wise search to generate frequent item sets. First it generates with 1-item sets, recursively generates 2-item set and then frequent 3-item set and continues until all the frequent item sets are generated. Han et. al [1] proposed the frequent pattern tree (FP-tree) and FP-growth algorithm to mine frequent patterns without candidate generation. Apriori and FP-growth algorithms measures the occurrence frequency i.e., support. Recently, Tanbeer et. al [4] introduced a new problem of discovering Regular Patterns that follow a temporal regularity in their occurrence behavior. With the help of regularity measure at which pattern occurs in a database at a user given maximum interval is called a *regular* pattern. They proposed a tree based data-structure, called RP-tree[4] that captures user-given regularity threshold based information from database with two database scans. With the first scan, it creates an item header table called regular table (R-table) to store items with respective regularity and support. Then with the second scan, the RP-tree is constructed in support-descending order only for regular items in R-table. Hence, because of the two database scans, the efficiency of an RP-tree is greatly reduced in discovering regular patterns in a transactional databases.

Therefore, in this paper, we propose a new method called Vertical Data Regular Patterns method (VDRP - method in short), using the same Transactional Database which is in [4] to mine regular patterns using vertical data format. By using Vertical Data Format [2, 5, 6], it will be able to judge whether the non-regular item sets before generating candidate item sets. The main idea of our new method is to develop a simple, but yet powerful, that captures the database content in full with one database scan to find regular items. The experimental results show the effectiveness of VDRP method in finding regular patterns in a Transactional Databases.

The rest of the paper is organized as follows. Section 2 summarizes the existing tree structure [4] to mine regular patterns. Section 3 introduces the problem definition of regular pattern mining. The method of VDRP to find regular patterns using vertical data format are given in section 4. Section 5, our experimental results are shown. Finally, we conclude the paper in section 6.

II. RELATED WORK

Mining regular patterns [4] with the help of user given regularity threshold at which a pattern occurs in a database, have been proposed as a tree-based data structure called RP-tree that captures user-given regularity threshold - based information from database with two database scans. It

creates an item header table, called regular table (R-table) to store items with respective regularity and support in the first scan. Then with the second scan the RP-tree is constructed in support-descending order only for regular items in R-table. Each entry in R-table consists of four fields in sequence (i, s, t_l, r); item name (i), support (s), *tid* of the last transaction where i occurred (t_l), and the regularity of i(r). After R-table is built, they generated R by removing all irregular items and arranging the items in support-descending order to facilitate the RP-tree construction. Using FP-tree construction technique [1], the RP-tree is constructed in such a way that, it only contains items in R in R-table order. No node in an RP-tree does maintain the support count field.

However, the transaction occurrence information is explicitly kept in a list called tid-list in the last node (say, tail-node) of the transaction. For simplicity of figures, the node traversal pointers in trees are not shown however they are maintained in a fashion like FP-tree does. They recursively mine the RP-tree of decreasing size to generate regular patterns by creating pattern-bases and corresponding conditional trees.

III. PROBLEM DEFINITION

Let L = $\{i_1, i_2, \ldots, i_n\}$ be a set of items. A set $X \subseteq L$ is called a *pattern* (or an itemset). A transaction t = (tid, Y) is a couple where tid is the transaction-id and Y is a pattern. A transactional database DB is a set of transactions $T = \{t_l, \ldots, t_m\}$ with m = |DB|, i.e., total number of transactions in DB. If $X \subseteq Y$, it is said that X occurs in t and denoted as t_j^X, $j \in$ [l, m]. Thus, $T^X = \{t_j^X, \ldots, t_k^X\}$, j ≤ k and j, k ∈ [l, m] is the set of all transactions where pattern X occurs. Let t^X_{j+1} and t_j^X, are two consecutive transactions in T^X. Then $p^X = t^X_{j+1} - t^X$, j ∈ [l, (m-l)] is a period of X and $P^X = \{p_1^X, \ldots, p_r^X\}$ is the set of all periods of X in DB. For simplicity, we consider the first and the last transactions in DB as 'null' with $t_{first} = 0$ and $t_{last} = t_m$ respectively. Let the max_period of X = $\text{Max}(t_{j+1}^X - t_j^X)$, j ∈ [l, (m-l)] be the largest period in P^X. We take max_period as the regular measure for a pattern and denote as *R* for X.

Therefore, a pattern is called a regular pattern if its regularity is no more than a user-given maximum regularity threshold called max_reg λ, with 1 ≤ λ ≤ |DB|. Regular pattern mining problem, given a λ and a DB, is to discover the complete set of regular patterns having regularity no more than λ in the DB.

IV. MINING REGULAR PATTERNS

First, scan the horizontal database (Table 1) into Vertical Database (Table 2) i.e., {*item* : *TID*_ set} where item is an item name and TID_set is the set of transaction identifiers containing the item. The regular patterns are the patterns that are less than or equal to user given regularity threshold i.e., (λ = 3).

Table I. Transactional Database

TID	Itemsets
1	a, d
2	b, c, a, e
3	a, e, b, f
4	a, e, b, c
5	a, b, e, f
6	b, c, d
7	c, e, d
8	d, e, f
9	d, c, b

VDRP – method:

Input : DB, λ = 3

Output : Complete regular Patterns

Procedure :

Let $X_i \subseteq$ L be a k-itemset
$P^X_i = 0$ for all X_i
For each X_i
Find the next transaction T_j
$P^X_i = j - P^X_i$
Max_reg (R) = max(P^X_i)
repeat
If max_reg > λ
Delete the itemset
Else
X_i is a regular itemset
Increase the k value using 'and operation' until no candidate is generated.

Table II. Vertical Data Format with P^X and R

Itemset	TID-Set	P^X	R
a	1, 2, 3, 4, 5	1, 1, 1, 1, 1, 4	4
b	2, 3, 4, 5, 6, 9	2, 1, 1, 1, 1, 3	3
c	2, 4, 6, 7, 9	2, 2, 2, 1, 2	2
d	1, 6, 7, 8, 9	1, 5, 1, 1, 1	5
e	2, 3, 4, 5, 7, 8	2, 1, 1, 1, 2, 1, 1	2
f	3, 5, 8	3, 2, 3, 1	3

The procedure is as follows - After getting Vertical Database format, find the P^X values of each itemset by substracting the TID-set values assuming the first transactions as $t_{first} = 0$ and $t_{last} = tm$. Then obtain the R value from P^X i.e., maximum value in P^X of an itemset. In our transaction DB *a* and *d* are deleted in Table 3 because R value is greater than user given regularity threshold i.e. λ = 3. Now use 'and operation' on Table 3 to get (k + 1) regular itemset i.e., in Table 4. We will stop doing 'and operation' until no regular items found. We shown only the regular patterns in VDRP – table. All other patterns are deleted because they are greater than our user given regularity threshold.

Table III. VDRP - Table

Itemset	TID-Set	P^X	R
b	2, 3, 4, 5, 6, 9	2, 1, 1, 1, 1, 3	3
c	2, 4, 6, 7, 9	2, 2, 2, 1, 2	2
e	2, 3, 4, 5, 7, 8	2, 1, 1, 1, 2, 1, 1	2
f	3, 5, 8	3, 2, 3, 1	3

By using vertical database format there are various advantages such as it needs the original database scan only once, it needs simple operations like union, intersection, subtraction, delete etc., it reduces i/o operations since no read/write operations required and also no need of using any type of data structure like array, linked list etc.,

Table IV. VDRP – Table

Itemset	TID-Set	P^X	R
(b, c)	2, 4, 6, 9	2, 2, 2, 3	3
(c, e)	2, 4, 7	2, 2, 3, 2	3
(e, f)	3, 5, 8	3, 2, 3, 1	3

V. EXPERIMENT RESULTS

We compare VDRP performance with that of RP-tree over several synthetic data and real datasets (e.g., mushroom, chess etc.) which are commonly used in frequent pattern mining experiments since such datasets maintain the characteristics of transactional database and obtain from UCI Machine Learning Repository (University of California – Irvine, CA). Due to the space constraint we only report the results on a subset of them. All programs are written in Sun Microsystems Java and run with Windows XP on a 2.66 GHz machine with 1 GB of main memory. We observed the execution time for our VDRP method over RP – tree on different data sets. Figure 2 and 3 shows the execution time our method takes very less time compare to RP – tree execution time.

VI. CONCLUSION

In this paper we presented a VDRP method. This method is better than the existing RP – tree algorithm because it utilizes the advantages of Vertical Transaction

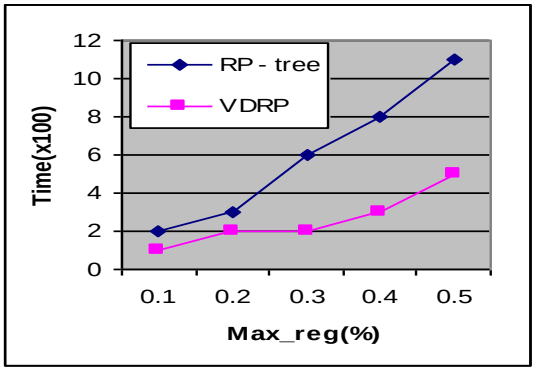

Figure.1 Execution time over mushroom

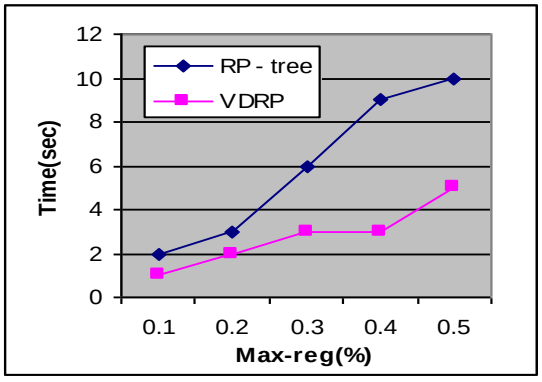

Figure.2 Execution time over T1014d100K

Database format and require only one database scan. This table (method) is efficient and scalable over large databases and faster than the RP-table. This method is very simple and works without complicated data structures. It needs only simple operations like union, intersection, subtraction etc., When regular k-item set to generate regular (k+1)-item set, the mode of intersection of any two sets is used. Pruning is done first in this paper, namely finding max_reg (R). If R is greater than user-given regularity threshold (λ), we'll delete the corresponding itemset. The experimental results shows that our VDRP can provide the time and memory efficiency during the regular pattern mining.

VII. REFERENCES

[1] Han, J., Yin, Y. Yin, "Mining Frequent Patterns without candidate generation", In Proc. ACM SIGMOD international Conference on management of Data, PP. 1-12 (2000).

[2] Jiawei Han, Micheline Kamber, "Data Mining : Concepts and Techniques", 2nd ed. An Imprint of Elsevier, Morgan Kaufmann publishers, pp. 232-248, 2006.

[3] R. Agarwal, and R. Srikanth, "Fast algorithms for mining association rules", In Proc. 1994 Int. Conf. Very Large Databases (VLDBA'94), pages 487- 499, Santiago, Chile, Sept. 1994.

[4] S. K. Tanbeer, C. F. Ahmed, B.S. Jeong, and Y.K. Lee, "Mining Regular Patterns in Transactional Databases", IEICE Trans. On Information Systems, E91-D, 11, pp. 2568-2577, 2008.

[5] G. Yi-ming, W. Zhi-jun, "A Vertical format algorithm for mining frequent item sets", IEEE Transactions, pp. 11-13, 2010.

[6] Mohammed J. Zaki, karam Gouda. "Fast Vertical Mining using Diffsets", SIGKDD '03, August 24 - 27, 2003, Copyright 2003 ACM 1-58113-737-0/03/0008.