Engineering Sciences International Research Journal
ISSN (Print) : 2330 – 4338 ; Volume 1 Issue 1 ǁ February. 2013 ǁ

Improved Mining Regular Pattern Method Over Apriori

M. Suneetha[1], M. Jyothi[2]

Abstract: The association rules are used to find interesting patterns from large collections of data which expresses an association between items or sets of items. Frequent item sets play an essential role in association rule mining. The closed frequent item set mining is advancement to frequent item set mining for association rules, which become an interesting topic for Researchers. In this paper we present study on extracting association rules that considers the number of database scans, and the interestingness of the rules. Finding regular patterns in a transactional database by using vertical data format with one database scan that removes the disadvantages of Apriori algorithm and is efficient in terms of number of database scans. In this method by using vertical data regular pattern -table for generating the complete set of regular patterns in a transactional database for a user given regularity threshold.

Keywords: Data Mining, Association Rule Mining Algorithms, Apriori Algorithm, Vertical databases, Regular patterns, Transactional database

1. INTRODUCTION

Frequent Pattern Mining 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. 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.

Frequent pattern tree (FP-tree) and FP-growth algorithm to mine frequent patterns without candidate generation. With the help of regularity measure at which pattern occurs in a database at a user given maximum interval is called a *regular* pattern.

In this paper, we present method [1]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.

2. ASSOCIATION RULE MINING

Association rule mining is to find out association rules that satisfy the predefined minimum support and confidence from a given database. The problem is usually decomposed into two sub problems. One is to find those item sets whose occurrences exceed a predefined threshold in the database; those item sets are called frequent or large item sets. The second problem is to generate association rules from those large item sets with the constraints of minimal confidence.

Apriori Algorithm

Apriori Algorithm can be used to generate all frequent item set. A Frequent item set is an item set whose support is greater than some user-specified minimum support (denoted L_k, where k is the size of the item set). A Candidate item set is a potentially frequent item set (denoted C_k, where k is the size of the item set).

1. Pass1

1. Generate the candidate itemsets in C_1
2. Save the frequent itemsets in L_1

2. Pass k

(i). Generate the candidate itemsets in C_k from the frequent itemsets in L_{k-1} Join $L_{k-1}p$ with $L_{k-1}q$, as follows:
insert into C_k select p.item1, p.item2, . . . , p.itemk-1, q.itemk-1 from $L_{k-1}p$, $L_{k-1}q$

where, p.item1 = q.item1, . . . p.itemk-2 = q.itemk-2, p.itemk-1<q. itemk-1

- Generate all (k-1)-subsets from the candidate itemsets in C_k
- Prune all candidate itemsets from C_k where, some (k-1)-subset of the candidate itemset is not in the frequent itemset L_{k-1}

(ii). Scan the transaction database to determine the support for each candidate itemset in C_k

(iii). Save the frequent itemsets in L_k.

Limitations of APRIORI Algorithm

Apriori algorithm, in spite of being simple and clear, has some limitation. It is costly to handle a huge number of candidate sets. Apriori Algorithm Scans the database too many times, When the database storing a large number of data services, the limited memory capacity, the system I/O load, considerable time scanning the database will be a very long time, so efficiency is very low.

3. VERTICAL DATA REGULAR PATTERN METHOD

Let L = {*i1, i2, …,in*} 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* = {*tl, …, tm*} 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 *tjX*, $j \in$ [l, m]. Thus, *T X* = {tjX, …,tkX}, j≤ k and j, k $\in$ [l, m] is the set of all transactions where pattern X occurs. Let $t^X j+1$ and tj^X, are two consecutive transactions in T^X. Then $p^X = t^X j+1 – t^X$, j $\in$ [l, (m-l)] is a period of X and $P^X = \{p1^X, …, pr^X\}$ 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 = Max($tj+1^X – tj^X$), j $\in$ [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 and a DB, is to discover the complete set of regular patterns having regularity no more than λ in the DB.

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 1 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 item set

Increase the k value using "and" operation until no candidate is generated.

Table 2 Vertical Data Format with PX 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 3 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 4 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

4. CONCLUSION

VDRP method is better than the existing Apriori algorithm because it utilizes the advantages of Vertical

Transaction. Database format and require only one database scan. This table (method) is efficient and scalable over large databases

5. 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

[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.

[1]M.Suneetha received B.Tech degree in Computer Science and Information Technology from AITAM College, Tekkali, India and M.Tech degree in Software Engineering from JNTUH, Hyderabad, India. Currently she is working as a Assistant professor in Information Technology Department at GMRIT, Rajam. Her area of interest is Data Mining.
sunita.merugula@gmail.com

[2]M.Jyothi received B.Tech degree in Computer Science and Information Technology from AITAM College, Tekkali, India and M.Tech degree in Computer Science and Engineering from JNTUK, Kakinada, India. Currently she is working as a Assistant professor in Information Technology Department at GMRIT, Rajam. Her area of interest Information Security and Data Mining.
jyothirajb4u@gmail.com