Volume 1, Issue 9, December 2016 ISSN: 2456-0006

International Journal of Science Technology Management and Research

Available online at: www.ijstmr.com

An Efficient Approach for Mining Frequent Pattern with Time over Large Database

Chinamy Jain[1]
[1]CSE. Dept. M. Tech. IV Sem.
Lord Krishna College of Technology Indore
M.P, India

Vijay Kumar Verma[2]
[2]Asst. Prof. CSE Dept.
Lord Krishna College of Technology Indore
M.P India

Abstract: - The process of exploring and analyzing data from different perspective, using automatic or semiautomatic techniques is called Data mining. Data mining extracts knowledge or useful information and discovers correlations or meaningful patterns and rules from large databases. Using these patterns and rules it is possible for business enterprises to identify new and unexpected trends, subtle relations in the data and use them to increase revenue and cut cost. In this paper we proposed an efficient approach for mining frequent pattern based on time

Keywords: Progressive, Partition, Miner, Weighted, comparative

I. INTRODUCTION

The traditional data mining techniques not have ability to analyze variation of data over time and treat them as ordinary data. Temporal datasets includes stock market data, manufacturing or production data, maintenance data, web mining and point-of-sale records. Temporal data mining means mining or discovering knowledge and patterns from temporal databases. Temporal data mining is an extension of data mining with ability to include time attribute analysis. Due to the significance and complexity of the time attribute, a lot of different kinds of patterns are of interest .

Two time aspects are included in temporal databases namely, valid time and transaction time. The time period during which a fact is true with respect to the real world is considered as valid time and the time period during which a fact is stored in the database is called transaction time. According to these two time aspects temporal databases allow the division of three different forms . They are

1. A historical database stores data with respect to valid time.
2. A rollback database stores data with respect to transaction time.
3. A temporal database stores data with respect to both valid and transaction time, that is, they store the history of data with respect to valid time and transaction time. The medicine that has been attached on the packages of the medicine. Status of the products will be notified on the screen, just a few seconds later. The data has been taken from National Pharmaceutical Control Bureau Ministry of Health Malaysia.

II. TEMPORAL DATA MINING TASKS

A main question is how to apply traditional data mining techniques on a temporal database. Temporal data mining may involve the following areas of investigation. Temporal data mining tasks includes:

1. Temporal association rules
2. Temporal data classification and comparison
3. Temporal pattern analysis

4. Temporal clustering analysis
5. Temporal prediction and trend analysis
6. Temporal classification

III. RELATED WORK

A temporal association rule is defined as the frequency of an itemset over a time period T and is the number of transactions in which it occurs divided by total number of transaction over a time period. To solve the problem on handling time-series by including time expression into association rules temporal association rule mining has been introduced. Temporal association rule mining is first introduced by Wang, Yang and Muntz in years 1999-2001.Temporal association rule mining is introduced together with the introduction of the TAR (Temporal Association Rule) algorithm. With the help of Temporal association we can finds the valuable relationship among the different item sets, in temporal database. There are several types of temporal association rules defined by various researcher e such as inter transaction rules, episode rules, trend dependencies, sequence association rules [6,7, 8].

In 2008 Roddick and Spiliopoulou have presented a comprehensive overview of techniques for the mining of temporal data using three dimensions: data type, mining operations and type of timing information (ordering).

In 2009 Winarko and Roddick proposed a non Apriori-based technique that avoids multiple databases scans; these methods not only avoid multiple data scan but efficiently mine arrangements and rules in a temporal database. The main drawback of this method is that it do not consider any constraints for the temporal relations and do not examine any measures for their rules other than the traditional confidence.

In 2010 Tansel and Imberman proposed a method where association rules were extracted for consecutive time intervals with different time granularities. They proposed a simple operation that extracts portions of a temporal relation was used during mining process and was combined with the first step of discovering association rules. Using this approach, the process of knowledge discovery can observe the changes and variation in the association rules over the time period when these rules are valid.[10,11].

In 2010 Gharib et al. proposed a method for generating temporal association rules to solve the problem of handling time series by including time expressions into association rules. To solve this they extended an incremental algorithm to maintain the temporal association rules in a transaction database, at the same time maintains the benefits from the results of earlier mining to derive the final mining output.

In 2011 C. H. Lee et al proposed progressive partition miner (PPM). In PPM the database is first partitioned the dataset by the size of time granularity. Then It applies filtering threshold mechanism on partition of the database and prune out infrequent 2-itemsets. PPM efficiently reduction extra scanning process. PPM works efficiently with temporal datasets. However, the limitation of this technique is its ability to deal with problems of incremental mining.

In 2012 Cheng. Y. Chang et al Segment Progressive Filter (SPF) was introduced after PPM. SPF is based on the Segmentation and progressive filtering. It was introduced. SPF first divides the database into certain imposed time granularity. It further segments the database based on their common starting and ending times. For each part of the database it finds the 2-candidate item set with appropriate filtering threshold. After generating all candidates it generates the sub-candidate and counts for the value of support. Temporal databases are continuously updated or appended.

In 2013 Ru Miao et al presented the idea of Apriori-extended mining periodic temporal association rules (MPTAR). MPTAR solved this problem, by considering the exhibition period of individual item. MPTAR is also a two-step periodic rule mining method. The first step is mining the trend of continues attribute through cycle curve and the second step is calculating the period of the attribute. MPTAR did not define the cumulative threshold, and it is short of embracing upcoming transaction entries in the association rule mining.

IV. BASIC CONCEPTS

Temporal association rule adds time constraint (it can be time point or time range) on association rule. A transaction with time information can be described as: {TID, I1, I2 …In, Ts, Te}. TID is the ID for each transaction; n-itemsets means there are n items in the itemset; Ts and Te represent the start and the end of valid time respectively (or the start and the end of transaction). Valid time means the event occurring time, while transaction time the database time. Ts may equal Te, such as sale records in the supermarket (the transaction occurs at one moment). According to the definition of strong association rule "association rule strictly satisfies minimum support threshold and minimum confidence threshold", we can give the definition of strong temporal association rule.

Let min_s and min_c represent minimum support threshold and minimum confidence threshold respectively, if and only if during [ts, te], support ≥ min_s, confidence ≥ min_c, rule X→Y is a temporal association rule, which could be described as X→Y (support, confidence, [ts, te]).

V. PROGRESSIVE PARTITION MINER (PPM)

Basic steps used in PPM are

1. Find candidate 2-itemsets from transactional database.
2. Compare threshold with minimum support counts using time.
3. Stores frequent 2-itemsets from transactional database.
4. Find candidate 2-itemsets for the next partition add the previous frequent 2-itemsets and compare threshold with minimum support counts using time. Find frequent 2-itemsets for both partitions.
5. Repeat this process for all partitions

Consider a simple transaction database

<table>
<tr><td colspan="5">Transactional database</td></tr>
<tr><td rowspan="9">DB</td><td rowspan="5">P1</td><td>Date</td><td>TID</td><td>Transactions</td></tr>
<tr><td rowspan="4">Jan 01</td><td>T1</td><td>B, D</td><td rowspan="12">db 1,3</td></tr>
<tr><td>T2</td><td>B, C ,D</td></tr>
<tr><td>T3</td><td>B, C</td></tr>
<tr><td>T4</td><td>A, D</td></tr>
<tr><td rowspan="4">P2</td><td rowspan="4">Feb 01</td><td>T5</td><td>B, C ,E</td><td rowspan="8">db 2,3</td></tr>
<tr><td>T6</td><td>D, E</td></tr>
<tr><td>T7</td><td>A, B, C</td></tr>
<tr><td>T8</td><td>C, D, E</td></tr>
<tr><td rowspan="4">Db(Increment</td><td rowspan="4">P3</td><td rowspan="4">Mar 01</td><td>T9</td><td>B, C, E, F</td><td rowspan="4">db 3,3</td></tr>
<tr><td>T10</td><td>B, F</td></tr>
<tr><td>T11</td><td>A, D</td></tr>
<tr><td>T12</td><td>B, D, F</td></tr>
</table>

Table 1 Simple transaction database

Let minimum is support 30% now generate two item set for P1

P1		
C2	Start	Count

AD	1	1
BC	1	2
BD	1	2
CD	1	1

Table 2 candidate item set p1

Now check candidate item set for P1 minimum support 4*0.3=1.2

P1		
C2	Start	Count
BC	1	2
BD	1	2

Table 3 frequent item set for P1

Generate two item set for P1+P2

P2		
C2	Start	Count
AB	2	1
AC	2	1
BC	2	2
BE	2	1
CD	2	1
CE	2	2
DE	2	2

P1		
C2	Start	Count
BC	1	2
BD	1	2

⇩

P1+P2		
C2	Start	Count
AB	2	1
AC	2	1
BC	1	4
BD	1	2
BE	2	1
CD	2	1
CE	2	2
DE	2	2

Table 4 candidate item set for P1+P2

Support count for P1+P2= (4+4)*.03 = 2.4

P1+P2		
C2	Start	Count
BC	1	4
CE	2	2
DE	2	2

Table 5 frequent item set for P1+P2

Now generate candidate item set for P3

P3		
C2	Start	Count
AD	3	1
BC	3	1
BD	3	1

BE	3	1
BF	3	3
CE	3	1
CF	3	1
DF	3	1
EF	3	1

+

P1+P2		
C2	Start	Count
BC	1	4
CE	2	2
DE	2	2

⇩

P1+P2+P3		
C2	Start	Count
AD	3	1
BC	1	5
BD	3	1
BE	3	1
BF	3	3
CE	2	3
CF	3	1
DE	2	2
DF	3	1
EF	3	1

Table 6 candidate item set for P1+P2+P3

Minimum support for P1+P2+P3 = (4+4+4)*0.3=3.6

P1+P2+P3		
C2	Start	Count
BC	1	5
BF	3	3
CE	2	3

Table 7 frequent item set for P1+P2+P3

General temporal itemset in DB+db

Itemset	Start	End	Start	End	TI
	B		C		
BC	1	3	1	3	BC1,3
	B		F		
BF	1	3	3	3	BF3,3
	C		E		
CE	1	3	2	3	CE

Table 8 frequent pattern with start and end pattern

General sub temporal item set in DB+db

TI	SI
BC1,3	B1,3
	C 1,3
BF3,3	B 3,3
	F 3,3
CE 2,3	C 2,3

	E 2,3

Table 9 General sub temporal item

VI. PROPOSED APPROACH

In proposed approach each transaction period is first reflected by a proper weight assigned. Then time-variant database in light of weight periods of transactions and performs weight mining. Proposed approach explores newly defined concepts of weight support. Instead of using the traditional support threshold we used weight minimum support, denoted by min_Sw = { Σ |Pi| × W (Pi) } × min_supp, is employed for the mining of weight frequent pattern and corresponding weight values by a weight function W (•) in the weight period Pi of the database D. Finally we can generate same frequent item set which generated by the PPM by using simple weight function. Proposed approach is one straightforward approach to addressing the above issues is to employ the item-constraints and/or multiple supports strategies new coming items have higher weights for their item occurrences.

VII. EXPERIMENTAL RESULTS

For the experimental analysis, we execute proposed algorithm with PPM for 25 items with 10,000 transaction .Form the graph it is clear that proposed approach take less execution time as compared to PPM. When the new item came for particular time interval otherwise it work same as PPM .One more thing that are consider with the proposed approach we have to check the one item set of the previous time interval to find new item set for current scanning time interval

Minimum Support	Proposed	PPM
0.2	20	23
0.3	14	17
0.4	9	10

Table 10 comparison table

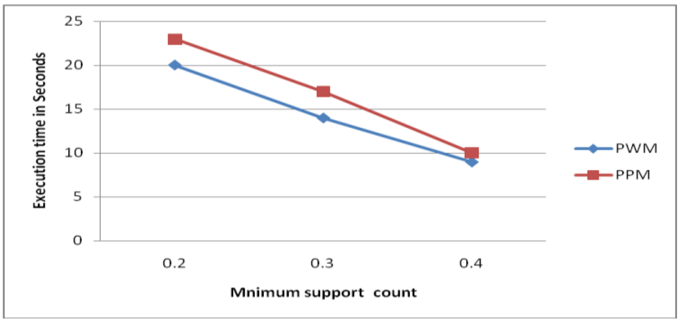

Figure 2 comparison graph

CONCLUSION

PPM focus on successive partition and calculate frequent item set and PWM assign weight to the partition in successive manner if new item are appears important to note that if we adopt single min_supp = 30% by Apriori, then the itemset {BF} will not be large since its occurrence in this transaction database is 3 which is smaller than min_ST = [12×0.3] = 4. However, itemset {BF} appears very frequently in the most recent partition of the database of which the weight is relatively large, thus discovering more desirable information. It can be seen that the algorithm Apriori is not able to discover the information behind the new coming data in the transaction database.

REFERENCE

1) J. Han, M. Kamber, Data mining, Concepts and techniques, Academic Press, 2003.
2) Arun K. Pujari, Data mining Techniques, University Press (India) Private Limited 2006.
3) D. Hand, H. Mannila, P. Smyth, Principles of Data Mining, Prentice Hall of India, 2004.
4) Principles of Data Mining by David Hand, Heikki Mannila and Padhraic Smyth ISBN: 026208290x The MIT Press © 2001
5) N. Pughazendi, Dr. M. Punithavalli, "Temporal databases and frequent pattern mining techniques", International Journal of P2P Network Trends and Technology July to Aug Issue 2011
6) Weiqiang Lin, Mehmet A. Orgun, Graham J. Williams, "An Overview of Temporal Data Mining", The Australasian Data Mining Workshop.
7) Mohsin Naqvi, Kashif Hussain, Sohail Asghar, Simon Fong, "Mining Temporal Association Rules with Incremental Standing for Segment Progressive Filter"
8) Tarek F. Gharib, Hamed Nassar, Mohamed Taha, Ajith Abraham "An efficient algorithm for incremental mining of temporal association rules", 2010 Elsevier B.V. All rights reserved
9) Chang-Hung Lee, Cheng-Ru Lin, and Ming-Syan Chen, "On Mining General Temporal Association Rules in a Publication Database"
10) ZHAI Lianga, TANG Xinming, LI Lina, JIANG Wenliang, "Temporal Association Rule Mining Based On T-Apriori Algorithm And Its Typical Application"
11) Chang-Hung Lee1, Jian Chih Ou, and Ming-Syan Chen, "Progressive Weighted Miner: An Efficient Method for Time-Constraint Mining"
12) Anour F.A., Dafa-Alla, Ho Sun Shon, Khalid E.K. Saeed, Minghao Piao, Un-il Yun, Kyung Joo Cheoi and Keun Ho Ryu "IMTAR: Incremental Mining of General Temporal Association Rules", Journal of Information Processing Systems, Vol.6, No.2, June 2010
13) Sotiris Kotsiantis, Dimitris Kanellopoulos, "Association Rules Mining: A Recent Overview GESTS", International Transactions on Computer Science and Engineering, Vol.32 (1), 2006, pp. 71-82
14) Litvak Marina, "Temporal Mining Algorithms: Generalization and Performance Improvements", The research work for this thesis has been carried out at Ben-Gurion University of the Negev under the supervision of Prof. Ehud Gudes November 2004
15) D.sujatha, Prof. B. Deekshatulu, "Algorithm for mining time varying frequent itemsets", Journal of theoretical and applied information technology 2005 - 2009.
16) Tannu Arora1, Rahul Yadav2, " Improved Association Mining Algorithm for Large Dataset", IJCEM International Journal of Computational Engineering & Management, Vol. 13, July 2011
17) Mamta Dhanda, Sonali Guglani, Gaurav Gupta, "Mining Efficient Association Rules Through Apriori Algorithm Using Attributes", IJCST Vol. 2, Issue 3, September 2011
18) Chin-Chen Chang , Yu-Chiang Li ,Jung-San Lee, " An Efficient Algorithm for Incremental Mining of Association Rules", Proceedings of the 15th International Workshop on Research Issues in Data Engineering: Stream Data Mining and Applications (RIDE-SDMA'05) © 2005 IEEE
19) N.L. Sarda, N. V. Srinivas, "An Adaptive Algorithm for Incremental Mining of Association Rules", Authorized licensed use limited to: Indian Institute of Technology Bombay. Downloaded on April 24, 2009 at IEEE Explore.
20) Sotiris Kotsiantis, Dimitris Kanellopoulos, "Association Rules Mining: A Recent Overview GESTS", International Transactions on Computer Science and Engineering, Vol.32 (1), 2006, pp. 71-82
21) Sandhya Rani Jetti, Sujatha D, "Mining Frequent Item Sets from incremental database:A single pass approach", International Journal of Scientific & Engineering Research, Volume 2, Issue 12, December-2011 1 ISSN 2229-5518
22) Jyoti Jadhav, Lata Ragha, Vijay Katkar Incremental Frequent Pattern Mining International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-1, Issue-6, August 2012
23) Ratchadaporn Amornchewin , Worapoj Kreesuradej, "Incremental Association Rule Mining Using Promising Frequent Itemset Algorithm", 1-4244-0983 2007 IEEE
24) Sheila A. Abaya "Association Rule Mining based on Apriori Algorithm in Minimizing Candidate Generation", International Journal of Scientific & Engineering Research Volume 3, Issue 7, July-2012 1 ISSN 2229-5518
25) Deepak Garg, Hemant Sharma, "Comparative Analysis of Various Approaches Used in Frequent Pattern Mining", (IJACSA) International Journal of Advanced Computer Science and Applications, Special Issue on Artificial Intelligence pattern Mining.