International Journal of Recent Research and Review, Vol.VII, Issue 2, June 2014

ISSN 2277 – 8322

Cost Vector Matrix – A New Approach to Association Rule Mining

Shikhar Kumar Jain[1], Dr. Meenu Dave[2], Ankur Agrawal[3]

[1,3]*M.Tech. Scholar, Department of Computer Science, Jagan Nath University, Jaipur ,India*

[2]*Principal, JaganNath Institute of Engineering & Technology, Jaipur, India*

Email: [1]shikharjain7254@hotmail.com, [2]meenu.s.dave@gmail.com, [3]4kankur@gmail.com

***Abstract*-Association Rule Mining is used to generate frequent datasets in transactional database. Many algorithms are used for the generation of frequent item-sets. In applications like frequent pattern analysis, usually market transaction, banking transaction, web log data, shopping mart data, etc. transactional data is required and here in this research work we have introduced a new algorithm for mining of frequent item-sets, known by the name "CVM Association Mining". In this approach we use cost matrix for items to generate new association rules. There are two main weakness of Apriori algorithm – first is the generation of large number of candidate item-sets and second is that database passes are equal to the length of frequent item-set. These two problems have been reduced through the explained algorithm.**

***Keywords*- Association Rule Mining, Cost Matrix, Cost Vector, CVM, Data Mining, Frequent Pattern**

I. INTRODUCTION

Association rule mining is one of the most important and well researched techniques of data mining [1]. Association rule mining finds frequent patterns, associations, correlations, or causal structures among sets of items or objects in transaction databases, relational databases, and other information repositories [2]. Association rules are widely used in various areas such as telecommunication networks, market and risk management, inventory control etc. [3].

The main aim of association rule mining is to provide better rule and frequent item-sets for prediction and decision making. It explains the core concept of frequent item-set and association rule mining. Frequent item-sets are those items which occur in transactions frequently. For example, a person who purchases Health Policy, generally purchases Accident Policy. Both these items frequently appear in database of an insurance company [4].

Section II of the research paper gives an overview of association rule mining, section III describes about the issues in association rule mining; section IV represents the CVM association rule mining technique. Implementation of CVM algorithm is described in section V, section VI provides the results of experimental analysis and the last section VII concludes the whole research.

II. OVERVIEW OF ASSOCIATION RULE MINING

A. Definition

Let I = {i1, i2, i3, i4……. im} be a set of different items. Let D be a set of transactions where each transaction T is a nonempty item-set such that $T \subseteq I$. Each transaction is associated with an identifier, called TID. Let P be a set of different items. Any transaction T has element P if and only if the element P is with that transaction T. An association rule is an implication of the form P => Q, where P, $Q \subseteq I$ and $P \cap Q = \emptyset$. Here P is called the antecedent and Q is called the consequent of the rule [5].

B. Concept of Database

Horizontal Database-A set of transactions in TID-item set format (that is, {TID: item set}), where TID is a transaction–id and item set is the set of items bought is transaction TID. This data format is known as horizontal data format [6].

Table I
Horizontal Database

Item	TID
T1	**A,B,C,D**
T2	**B,C**
T3	**A,B,C**
T4	**A,B,D**

Vertical Database- Database can also be presented in item-TID set format (that is {item: TID set}), where item is the set of items and TID is the set of transaction identifiers containing the items. This data format is known as vertical data format [6].

Table II
Vertical Database

Item	TID
A	**T1,T3,T4**
B	**T1,T2,T3,T4**
C	**T1,T2,T3**
D	**T1,T4**

Association rule can be generated by using vertical data format also. It is faster to generate the rules and is easy to implement. In this research work, we used vertical data format to generate the association rules.

C. Advantage of Vertical Database

(i) Computation of support count; only the intersection of the TID sets placed in it is carried out.

(ii) Only frequent item-sets are considered for further computations, so number of database scans is reduced at each level.

(iii) The vertical layout is more versatile in supporting various search techniques [6].

III. ISSUES IN ASSOCIATION RULE MINING

Large amount of transactions are produced on a daily basis in many application areas. This data is required to be processed for storage and analysis. Existing technologies are good but new and better ones are always needed.

Some issues in mining technology are discussed below-

1. More efficient methods for association rule mining need to be developed which make up for a balance between computation cost and communication cost.
2. Database-independent measurements should be established [7].
3. Techniques for mining association rules in multi-databases should be explored.
4. Efficient algorithms should be developed for data mining on XML databases, Web Usage Mining, social network analysis and mining for business functionalities.
5. Single scan and online mining methods should be developed to discover knowledge or patterns from data streams [7].
6. Deep-level association rules should be identified.

IV. CVM ASSOCIATION MINING

A. Encoded Database

The concept of encoded database is based on binary representation of vertical transactional database. For generation of frequent patterns we design a new approach of cost vector matrix. In this approach the transactional information of each item is converted into a binary code. The transaction which has the required item, assigns the code value "1" to TID and "0" otherwise. This binary encoding generates a code vector for each item for vertical database. We can say that here

we convert the TID to a binary code0 or 1, based on the presence or absence of an item in the transaction.

Table III
Cost Matrix for Transaction Database

Item	TID			
	T1	T2	T3	T4
A	1	0	1	1
B	1	1	1	1
C	1	1	1	0
D	1	0	0	1

B. Proposed Algorithm

To generate frequent patterns by using Cost Vector Matrix (CVM) association mining algorithm we use encoded database as cost vector matrix.

Step1: Scan the vertical database and assign the cost 1 or 0 for each item present or not, in cost matrix of size n*m. // where n= number of items, m= number of transactions.

Step2: To calculate the support count or frequency of each item, multiply the cost matrix C_{nm} with a unit vector matrix V_m of size m*1.

$$F_{sup} = C_{nm} * V_m$$

Step3: If any of F_{sup} value is less than minimum support then discard the item.

Step4: Generation of association rules on k+1 level or every next level

-Derive a mask matrix C_m. Eliminate the top row of existing matrix C_{nm} and intersect both.

Case-1 If all the derived subsets are of size n, then go to step 6.

Case-2 If current loop generates null subset, then terminate the loop for k^{th} level. Repeat step 4 for k+1 level.

Case-3 If k+2 item-sets are generated at k+1 level then discard the item set.

Case-4 If support count of k+1 item <min_sup, then discard the item.

Step5: To calculate the frequency of matrix, go to step 2.

Step6: Exit.

C. Flow Graph of CVM Algorithm

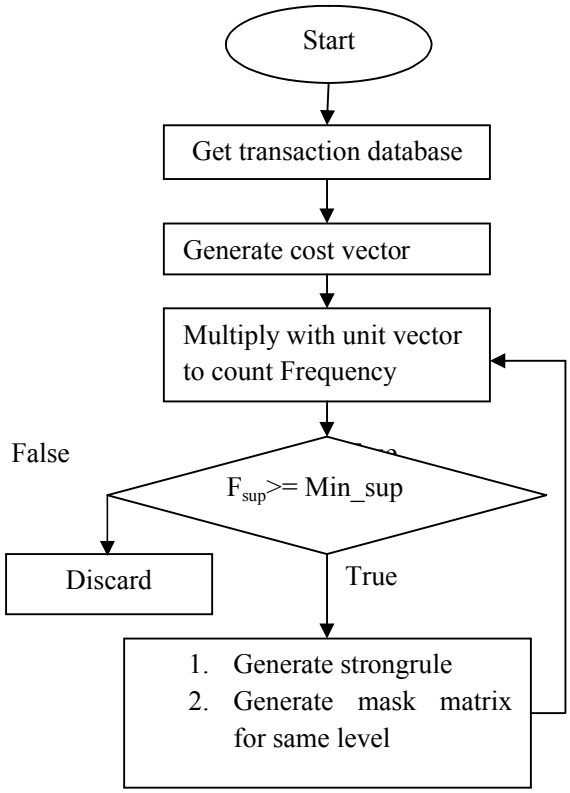

Fig. 1 Flow Graph of CVM Association Mining Algorithm

V. CVM ASSOCIATION MINING IMPLEMENTATION

To explain the implementation of CVM association mining algorithm we take an example of transactional database and generate frequent item-sets and finally the results are compared with the existing Apriori algorithm.

Let us consider a small transactional table with 7 transactions and 4 items as given in Table IV below. For this database we hereby generate cost matrix and their graph charts which represent the frequency of item-sets. Minimum support threshold is equal to 3.

Table IV
Transactional Database

Item	TID
A	**T1,T2,T4,T6,T7**
B	**T1,T2,T4,T5,T7**
C	**T1,T2,T3,T4,T5**
D	**T1,T2,T3,T5,T6,T7**

Fig. 2 shows the cost vector matrix for transactional database. Each item is represented as a cost vector in this matrix.

	T1	T2	T3	T4	T5	T6	T7
A	1	1	0	1	0	1	1
B	1	1	0	1	1	0	1
C	1	1	1	1	1	0	0
D	1	1	1	0	1	1	1

Fig. 2 Cost Vector Matrix for Table IV

The following graphs show the frequent patterns for different passes or different pattern lengths. Fig. 3 shows frequent patterns of length 2 and Fig. 4 shows frequent patterns of length 3.

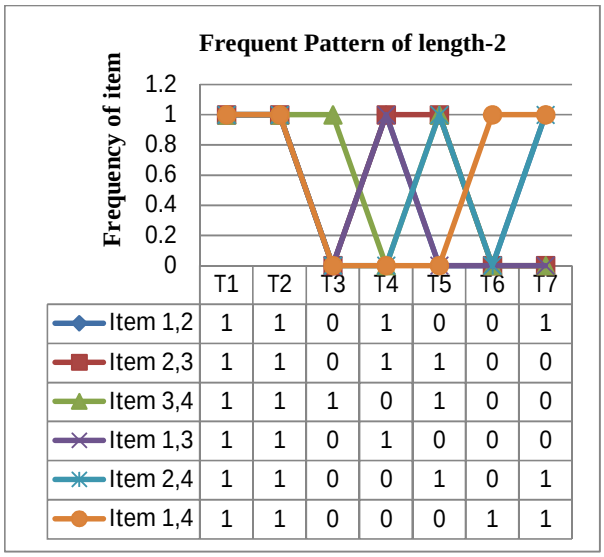

Fig. 3 Cost Vector Matrix for Pattern length-2

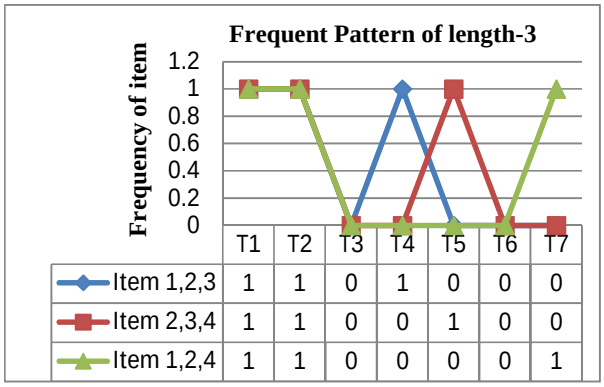

Fig. 4 Cost Vector Matrix for Pattern length-3

VI. EXPERIMENTAL RESULTS

Through this experiment, comparison of both Apriori and CVM algorithms based on number of database scans and memory locations required to store item-sets has been carried out.

Here, we consider four transactional databases with different number of items and transactions. The results are represented in Table V based on number of database scan and Table VI represents the results based on number of memory locations required.

Table V
Experimental Results for Database Scan

S. No.	No. of Items	No. of Transactions	No. of Database Scan	
			Apriori	CVM
1.	8	6	98	64
2.	5	9	22	18
3.	5	8	76	66
4.	4	7	50	41

Graph in Fig. 5 represents the comparison between Apriori algorithm and CVM association mining algorithm based on number of database scans for given data in Table V. X-axis of graph represents the value for no. of items, transaction pairs and Y-axis represents no. of database scans.

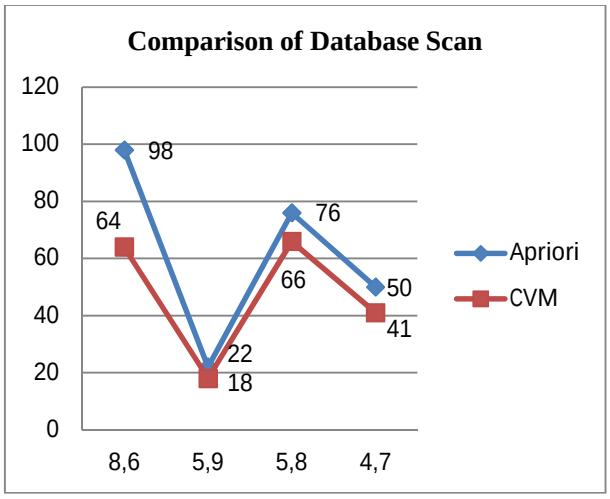

Fig. 5 Comparison between Apriori and CVM algorithms based on no. of database scans required

Graph in Fig. 6 represents the comparison between Apriori algorithm and CVM association mining algorithm based on number of memory location required for given data in Table VI. X-axis of graph represents the value for no. of items, transaction pairs and Y-axis represents no. of memory locations.

Table VI
Experimental Results for Memory Locations

S. No.	No. of Items	No. of Transactions	No. of Memory Location Required	
			Apriori	CVM
1.	8	6	74	25
2.	5	9	26	11
3.	5	8	64	28
4.	4	7	45	19

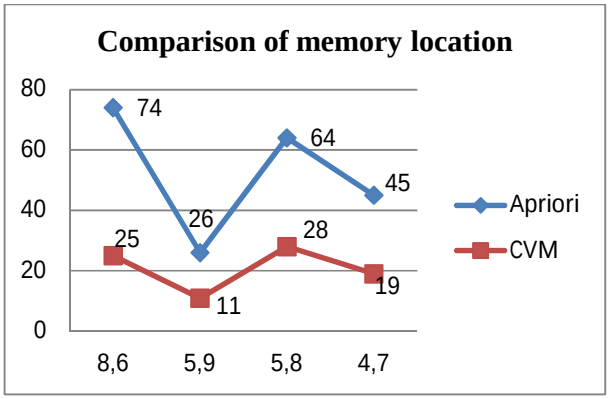

Fig. 6 Comparison between Apriori and CVM algorithms based on number of memory location required

VII. CONCLUSION

CVM association mining is a better approach as compared to the existing Apriori algorithm in terms of database scan, time and space complexity. In this approach there is no need to scan the database for support count. Analysis purely depends on the number of items and the number of transactions.

VIII. REFERENCES

[1] K. Lavanya, A. Kamala Priya, P. Suresh Babu, "An Overview of Association Rule Mining", *International Journal of Electronics Communication and Computer Engineering*, Volume 3, Issue 4, ISSN (Online) 2249 – 071X, ISSN (Print): 2278 – 4209, 2012.

[2] Bharat Gupta, "A Better Approach to Mine Frequent Itemsets using Apriori and FP-Tree Approach",M.E. thesis in CSE Department, Thapar University, Patiala, 2011.

[3] Sotiris Kotsiantis, Dimitris Kanellopoulos, "Association Rule Mining: A Recent Overview",*GESTS International Transactions on Computer Science and Engineering*, Vol.32 (1), pp. 71-82, 2006.

[4] Vimal Ghorecha, "Comparative Evaluation of Association Rule Mining Algorithms with Frequent Item Sets", *IOSR Journal of Computer Engineering*, Volume 9, e-ISSN: 2278-0661, p- ISSN: 2278-8727, PP 08-14, Issue 5 (Mar. - Apr. 2013).

[5] Jiawei Han, Micheline Kamber, Jian Pei, *Data Mining Concepts and Techniques*, 2nd ed.,Morgan Kaufmann Publishers, 225 Wyman Street, Waltham, USA, 2012.

[6] Sanat Jain, Swati Kabra, "Mining & Optimization of Association Rules Using Effective Algorithm", *International journal of emerging technology and advanced engineering,* volume 2, ISSN 2250-2459, issue 4, April 2012.

[7] Ziauddin, Shahid Kammal, Khaiuz Zaman Khan, Muhammad Ijaz Khan, "Research on Association Rule Mining", *Advances in Computational Mathematics and its Applications (ACMA)* Vol. 2, No.1, ISSN: 2167-6356, 2012.