An Efficient Approach for Association Rule Mining by using Two Level Compression Technique

Sheetal Bagde
CSE Deptt
BUIT Bhopal

Anju Singh
Deptt of CS and IT
UTD BU Bhopal

ABSTRACT
One of the most important techniques of data mining is Association rule mining. Association rule mining (ARM) is a mining technique to integrate data from large databases and to extract the interesting correlations, frequent patterns and associations among the huge collection of data. In this paper, we have proposed the two level compression techniques for finding frequent items from the two level compressed databases and generate association rules with minimum time.

Keywords
Association rule mining, Compressed database, Data mining, Frequent Patterns.

1. INTRODUCTION
Association rules are if and then statements that used to reveal relationships between uncorrelated data in a database, relational database or other information repository. It is used to extract the relationships between the objects data which are frequently used together. Applications of association rules are basket data analysis, storage planning etc. For example, if the customer buys milk then he may also buy bread. There are two significant measures that association rules uses, support and confidence. It describes the relationships and rules created by studying data for frequently used if and then patterns. Association rules are generally required to satisfy a user-defined minimum support and a user–defined minimum confidence[3].

Support**:** Support defines the transactions that contains itemset. If p, q are two itemsets, then the support can be defined as the transaction T which defines p / q.

Confidence: Confidence is defined as the percentage of transactions where the itemsets are most likely to occur. If p, q are two itemsets, then, the probability p U q is a subset of transaction, T is called as the confidence.

Frequent patterns algorithms are Apriori algorithm [1][2] Frequent pattern growth algorithm and Eclat algorithm. These algorithms are used to generates rules on associated attributes.

Apriori algorithm [4] is a two stage process. First, the candidate item set generation and second, the rule generation. Before starting the working procedure of apriori algorithm, the minimum support P is defined by user . Apriori algorithm starts by scanning the complete database, D and find all the frequent items from the database D . First scan the complete database only for 1-itemsets, and then successive iterations deals the 2-itemset. Thus new list of frequent items are created. The process continues untill all the frequent itemsets are extracted from D. Only those frequent items whose minimum support is greater than or equal to P is taken for rule generation.

2. LITERATURE SURVEY
Park J.S. et al. [5] developed an algorithm for mining association rules with adjustable accuracy. In this algorithm two methods for mining association rules with adjustable accuracy are developed. By using the sampling concept , both methods obtain some crucial knowledge from a sampled subset first, and knowledge perform efficient association rule mining on the whole database. In order to obtain desired level of accuracy, a technique of relaxing the support factor, based on sampling size was prepared.

Badri patel et al. [6] describes an Apriori algorithm and association rule mining and to improved algorithm by using the Ant colony optimization algorithm. ACO was introduced by dorigo and has developed substainly in the last few years. A huge amount of data has collected by many organization ,public sector and to store this dataset on the database systems. For analyzing the information system arise two major problems. First to reduce redundant objects and attributes so as to obtain the minimum subset of attributes ensuring a good estimation of classes and a good quality of classification. Second is representing the information system as a decision table which shows dependencies between the minimum subset of attributes and particular class numbers without redundancy. The working process of Apriori algorithm defines in steps. It is the two step processes used to find the frequent item set to join and prune. ACO algorithm was encouraged from original behaviour of ant colonies. It is used to solve to many hard optimizations including the travelling salesman problem. ACO system consists two rules. First, the local pheromone update rule, which is applied in constructing solution. Second one is global pheromone update rule which is applied in ant construction. ACO algorithm describes two more methods, namely trail evaporation and optionally deamonactions. ACO algorithm is used for the particular problem of minimizing the number of association rules. Apriori algorithm uses transaction data set and user defined support and confidence value then generates the association rules. These association rule set is distinct and continues. The weak rules are required to discard. Parag Deoskar [7] proposed an algorithm for detecting the lung cancer patients. This algorithm is based on ant colony optimization. It classify increase the lung cancer chances detect the early and correct decision which prove to be detect in battling disease.

Sunita Sarawagi et al. [8] researched several architectural options for integrating mining with RDBMS. Jacky et al. [9] analyzed how XQuery can be used to mine association rules from XML database. The essential adaptable structure and semantics of XML data bases makes more challenges [10].

Priyanka Asthana et. al. [11] describes an algorithm which create a cache database for each transaction using hash map. This cache copy is used to search for finding itemsets. The

algorithm has reduced the time complexity of apriori algorithm using cache database and produces more accurate results in less time.

Virkram Garg et. al.describes [12] for hiding the sensitive information by using the data distoration technique. It uses the concept of representative rule which is used to purne the number of association rules. It hides the more number of rules while making the minimum database scan. It also increases the memory utilization.

3. PROPOSED WORK

In this proposed work, uses a two level compressed database to find out the frequent itemsets and generates association rules.

This approach will work as follows:

A) We will scan the database only once and count the values of one‘s entries except zero’s from the dataset by using sparse matrix data storage format . and to create a compressed data format. In this sparse matrix storage format contains the values of rows , columns values and values of non zero elements .This is first level of compression of given dataset.

B) We arrange the no of rows and columns in rows and columns set.

C) We convert the rows set into column and columns sets into rows.

D) Finally, arrange the rows and columns indices into number of slots. These number of slots creates again compressed database for finding the frequent items and generates association rules.

This approach uses less memory space for storing the database and provides more accurate result with less in time.

3.1 Steps of the Algorithm

1. Given D , the database

2. Convert the given database into sparse matrix data structure. 3. Arrange the no of rows and column in the rows set and columns set..

4. Convert the value of rows set into column and columns set into row.

5. Arrange the rows and columns indices in the no. of slots.

6. Apply user threshold value (depends on user) .

7. Calculate the frequent item set Li for all i item set.

For i=1 to n do begin

if (support count of itemsets >=min_support)

Li =itemsets

else

discard the item set.

End

8. Frequent item set L= L1 union L2 union L3……..union Ln itemsets.

9. Apply confidence on the frequent itemsets and generate association rules.

The following Table 1. Shows the Contact lens dataset .The dataset denotes the following parameters are Age of person , Astigmatic ,Tear prod rate and contact lens

Table 1. Contact lens dataset

Age	**Astigmatic**	**Tear prod Rate**	**Contact lens**
Young	No	normal	soft
Young	Yes	reduced	none
Young	Yes	normal	hard
Pre presbyopic	No	reduced	none
Pre presbyopic	No	normal	soft
Pre presbyopic	Yes	normal	hard
Pre presbyopic	Yes	normal	none
Presbyopic	No	reduced	none
Presbyopic	No	normal	none
presbyopic	Yes	reduced	none
Presbyopic	Yes	normal	hard

The following table 2. Shows the sparse matrix of contact lens dataset. The dataset represents transaction list TID and the following attributes. Attributes are taken as follows Young=A,Prepresbyopic=B,Prebyopic=C,No_Astigmatic=D, Yes_Astigmatic,=E,Tear_Prod_normal=F,Tear_Prod_reduced

=G , Contact_lens _soft=H, Contact_lens_none=I Contact_lens_hard = J

Table 2. Sparse matrix of Contact lens dataset

TID	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**
T1	1	0	0	1	0	1	0	1	0
T2	1	0	0	0	1	0	1	0	1
T3	1	0	0	0	1	1	0	0	0
T4	0	1	0	1	0	0	1	0	1
T5	0	1	0	1	0	1	0	1	0
T6	0	1	0	0	1	1	0	0	0
T7	0	1	0	0	1	1	0	0	1
T8	0	0	1	1	0	0	1	0	1
T9	0	0	1	1	0	1	0	0	1
T10	0	0	1	0	1	0	1	0	1
T11	0	0	1	0	1	1	0	0	0

The following table 3. represents the sparse matrix data storage of contact lens dataset. Count the value of one's instead of zero's from table 2 and generates a compressed data structure by using sparse matrix storage format.

Table 3.Sparse matrix data storage of Contact lens

Rows value	Columns value	Values ones
T1	A	1
T1	D	1
T1	F	1
T1	H	1
T2	A	1
T2	E	1
T2	G	1
T2	I	1
T3	A	1
T3	E	1
T3	F	1
T3	J	1
T4	B	1
T4	D	1
T4	G	1
T4	I	1
T5	B	1
T5	D	1
T5	F	1
T5	H	1
T6	B	1
T6	E	1
T6	F	1
T6	J	1
T7	B	1
T7	E	1
T7	F	1
T7	I	1
T8	C	1
T8	D	1
T8	G	1
T8	I	1
T9	C	1
T9	D	1
T9	F	1
T9	I	1
T9	J	1
T10	C	1
T10	E	1
T10	G	1
T10	I	1
T11	C	1
T11	E	1

We will generate rows set and columns set from compressed dataset.

Rows set={ T1,T2,T3,T4,T5,T6,T7,T8,T9,T10,T11}

Columns set ={ A ,B,C,D ,E,F,G,H, I, J}

The table 4. shows conversion of rows set into column and column set into rows .

Table 4. Conversion of Rows set into column and Column set into row

S.no	Row	Column
1.	A	{T1,T2,T3}
2.	B	{T4,T5,T6 ,T7}
3.	C	{T8,T9,T10,T11}
4.	D	{T1,T4 ,T5,T8,T9}
5.	E	{ T2,T3,T6,T7,T10,T11}
6.	F	{ T1,T3,T5,T6,T7,T9,T11}
7.	G	{ T2,T4,T8,T10}
8.	H	{T1,T5}
10.	I	{ T2,T4,T7,T8,T9,T10}
11.	J	{ T3 ,T6,T9,T11}

Figure1. Show the pictorial representation of dataset of table 4. Arranged the no of rows and columns into number of slots and calculate the frequent itemsets from pictorial representation of dataset

.

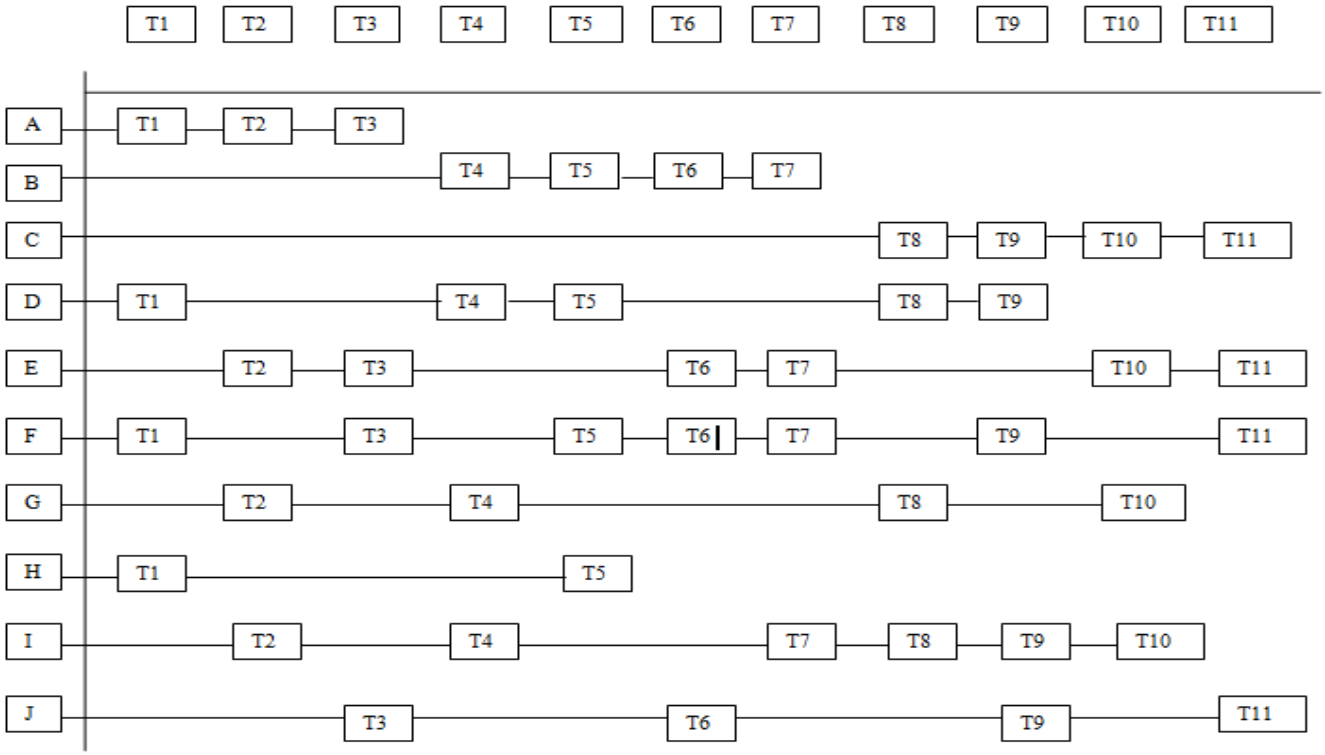

Figure 1. Pictorial Representation of dataset

For calculating frequent itemsets and we are taking support =(20%).If frequency of items is greater than or equal to support than the items is frequent otherwise the items is infrequent. Count the frequency of each items from the transaction and check corresponding items is present in other transaction. We take only those items whose frequency is satisfied the predefined support value.

For items n=1

L1={{A},{B},{C},{D},{E},{F},{G},{H},{I},{J}}

All are frequent itemset

For items n=2
L2={{A,E},{A,F},{B,D},{B,E},{B,F},{B,I},{C,D},{C,E}{C,F},{C,G},{C,I},{C,J},{D,F},{D,G}{D,H},{D,I},{E,F},{E,G},{E,I},{E,J},{F,H},{F,I},{F,J},{G,I}}

All are frequent itemsets

For items n=3
L3={{B,E,F}{C,D,I},{C,F,J},{C,G,I},{E,F,J},{E,G,I},{D,F,H},{D,G,I}}

All are frequent itemsets

Union of all the frequent items L= L1 U L2 U L3 U L4
L={{{B,E,F}{C,D,I},{C,F,J},{C,G,I},{E,F,J},{E,G,I},{D,F,H},{D,G,I}}U{{A,E}{A,F}{B,D},{B,E},{B,F},{B,I},{C,D},{C,E}{C,F},{C,G},{C,I},{C,J},{D,F}{D,G}{D,H},{D,I},{E,F},{E,G},{E,I},{E,J}{F,H},{F,I},{F,J},{G,I}}U{{A},{B},{C},{D},{E},{F},{G},{H},{I},{J}}U{NULL}}

L={{{B,E,F}{C,D,I},{C,F,J},{C,G,I}}{E,F,J},{E,G,I},{D,F,H},{D,G,I}}{{A,E}{A,F}{B,D},{B,E},{B,F},{B,I},{C,D},{C,E}{C,F},{C,G},{C,I},{C,J},{D,F}{D,G}{D,H},{D,I},{E,F},{E,G},{E,I},{E,J}{F,H},{F,I},{F,J},{G,I}}U{{A},{B},{C},{D},{E},{F},{G},{H},{I},{J}}}

4. EXPERIMENT AND RESULT ANALYSIS

The proposed algorithm has been developed in Matlab. The algorithm implemented on windows 8 64bits with i3 processor and 4 GB RAM.

These association rules are generated by proposed two level compression technique on contact lens dataset. Support = (20%) and when the confidence =(30%). We have mentioned here only few generated rules

1. No_Contact_lens => No_astigmatic

2. Tear_prod_reduced => No_Contact_lens

3.Contact_lens_hard=>Astigmatic,Tear_prod_normal

4. Contact_lens_hard => Astigmatic

5.No_astigmatic=>Tear_prod_reduced, No_Contact_lens

6.Tear_prod_reduced=>No_astigmatic,No_Cont ctlens

7.Astigmatic=>Tear_prod_normal,Contact_lens_hard

8.No_astigmatic=>Tearprod_reduced,No_Contactlens

9. Contact_lens_hard => Astigmatic

10.Tear_prod_normal,Contact_lens_hard=>Astigmatic

The following table 5. shows inference drawn from the generated rules.

Table 5. Inference drawn from generated rules

1.	If no contact lens than no astigmatic
2.	If tear prod is reduced than contact lens is not required
3.	If contact lens is hard than astigmatic , tear prod is normal
4.	If contact lens is hard than astigmatic
5.	If no astigmatic than tear prod is reduced and contact lens is not required
6.	If tear prod is reduced than no astigmatic and no contact lens is required
7.	If astigmatic than tear prod is normal and contact lens is hard.
8.	If no astigmatic than tear prod is reduced and no contact lens
9.	If contact lens is hard than astigmatic
10.	If tear prod is normal and contact lens is hard than astigmatic.

The following figure 2. represents comparison of the proposed two level compression technique , Apriori and FP growth algorithm for generating association rules with different support in (%) and when the confidence =30% in terms of execution time.

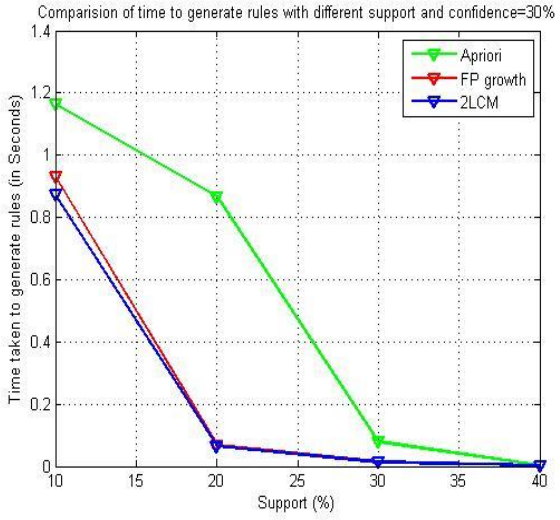

Figure 2. Comparison of the execution time of algorithms with different support (%) and when the confidence =30%.

In figure 3. represents the two different dataset on x axis and time (sec) is on Y axis. Comparison of 2LCM , Apriori and FP Growth algorithms for generating association rule with two different datasets.

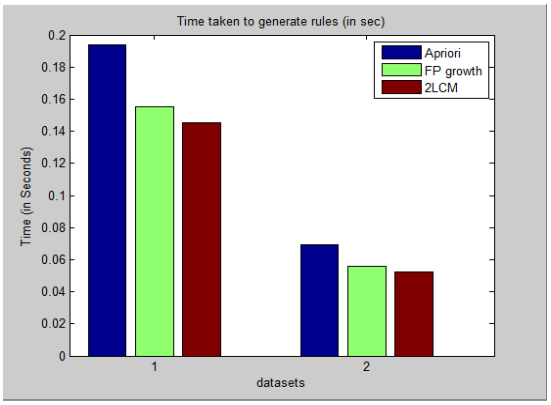

Figure 3. Comparison of 2LCM, Apriori And FP Growth algorithm for generating association rules with two different dataset.

In figure 4. shows the two dataset on X-axis and memory (in bits) on Y-axis. Applying 2LCM technique for compression on both dataset, and calculating memory space required for both datasets and this bar chart shows that if we are using 2LCM technique for compression then the resultant data takes less memory as compared to original data.

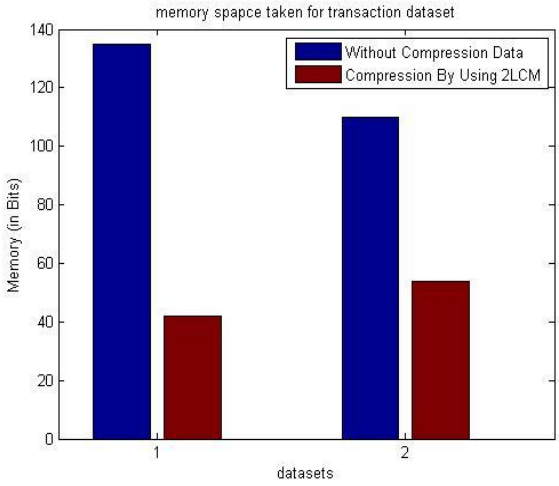

Figure 4. comparison of memory taken for compressed data and without compressed data

5. CONCLUSION

An efficient way for discovering the frequent set can be very useful in various data mining problems, such as discovery of association rules. In this dissertation work, new approaches to association rule mining has been explored in depth. Comparison of the algorithms, Apriori, FP growth and Proposed 2LCM Association Rule Mining was done in this work and there we found many advantages of Proposed 2 LCM Association Rule Mining over Apriori and FP growth.

It generates frequent itemsets and association rules in lesser time. It also uses less memory space. The result shows the effectiveness of our approach.

In future we will make my algorithm more easier to generate association rules. And will remove some steps by using new techniques for better performance. Here we will improve compressed data by using some other algorithms, because here we have seen that complexity is depends on scanning. If it have large dataset for scanning then time also will increase. Means if we use small or compressed dataset then time and

complexity will decrease. So for this point of view we have needed to reduce the dataset. And we know for compression we need more effort and algorithms.

In future, it may be possible that doesn't need compression for data and we can scan original data but not all entries. Algorithm only scans selected entries. Means concept of link list may be apply.

6. REFERENCES

[1] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," In Proc. of VLDB, pp. 487–499, 1994.

[2] L.Zeng Q He and Z shi "Parallel impleimplementation of apriori algorithm based on map reduce", In Proc. Of SNPD, pp.236-241, Aug 2012.

[3] H. Wu, Z. Lu, L. Pan, R. Xu, and W. Jiang, "An improved apriori-based algorithm for association rules mining," ,In Proc. of sixth international conference on fuzzy systems and knowledge discovery, pp. 51–55, 2009.

[4] L. Shi, J. niu Bai, and Y. lin Zhao, "Mining association rules based on apriori algorithm and application," In Proc. of IFCSTA, vol. 3, pp. 141– 145, Dec 2009.

[5] Park J.S, Phillip S.Y, Chen M,"Mining Association Rules with adjustable Accuracy", Conference on 0027sec Information and Knowledge Management In Proc. of the sixth international conference on Information and knowledge management, Las Vegas, Nevada, United States, pp 151 – 160, 1997.

[6] Badri patel ,Vijay K Chaudahri, Rajneesh K Karan,YK Rana,"Optimization of association rule mining apriori algorithm using Ant Colony optimization",International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Vol.1, Issue 1, March 2011

[7] Parag Deoskar, Divakar Singh, Anju Singh " An Efficient Support Based on Ant Colony Optimization Technique for Lung Cancer Data " ,International Journal of Advanced Research in Computer and Communication Engineering ,Vol. 2, Issue 9,September 2014..

[8] Sunita Sarawagi , Shiby Thomas and Rakesh Agrawal ,"Integrating Association rule mining with Relational Database Systems: Alternatives and Implications", ACM 089791-995-5/98/006, 1998.

[9] Jacky,W.W. Wan and Gillian Dobbie. "Mining Association rules from XML data using XQuery", Australian Computer Society, 2004

[10] Sasikala, D. and Premalatha, K. "Mining association rules from XML document using modified index table" , International Conference on Computer Communication and Informatics, Coimbatore, 4-6 Jan 2013

[11] Priyanka Asthana Divakar Singh," Improving Efficiency of Apriori Algorithm using Cache Database" ,International journal of Computer Application", Vol.75 , No. 13, 2013.

[12] Vikram Garg ,Anju Singh , Divakar Singh "A Hybrid Algorithm for Association Rule Hiding using Representative Rule", International Journal of Computer Application , Vol .97, No. 9 , July 2014.

[13] UCI machine repository