Survey of Various Frequent Pattern Mining Techniques

[1] Ms Priyanka Mali, [2] Prof. Shruti Yagnik

[1] Student of Gujarat Technological University, [2]Professor at LJ College Of Engineering & Technology
[1]Department of Computer Engineering,
[1] LJ College Of Engg. & Technology, Ahmedabad,India

***Abstract*— The basic frequent pattern mining algorithm is apriori. There are many improvements in frequent pattern mining algorithm one of them is hash based frequent itemset (HBFI) and another is double hash technique (DHBFI). In this paper we describe various technique review and comparative study for mining frequent patterns like apriori, direct hash technique, double hash technique.**

***Index Terms*— Frequent pattern mining, Double hash technique, Support**

1. INTRODUCTION:

. Data mining can be defined as knowledge discovery from data. The KDD process of data mining involves different steps like data cleaning, data integration, data selection, data transformation, mining of data, pattern evaluation and knowledge presentation identify frequent pattern then it will be helpful for market basket analysis, understanding disease behavior and predicting the market previously. Suppose the stoke of one store is analyzed to mine frequent pattern and they found milk and bread is frequently purchased together so the shopper can give discount on purchase of these items and can put them together to increase the sell. It will also be easy for customer to purchase items from same place. Similarly online database query from user is mined and frequent pattern is generated so we get idea about what kind of information users want to surf.

There are many data mining techniques like association rule mining, clustering, classification, outlier detection etc. Various application of data mining techniques are market analysis, medical analysis, business, bioinformatics and other areas which are beneficial for human's social and commercial activity like fraud detection, customer relationship management, ecommerce systems. Frequent pattern mining is useful to generate association rule. The bank database, different organizations database, web data, text data has been processed to generate frequent patterns and getting different interesting patterns for analysis which is beneficial in decision making purpose.

The frequency of some transaction in given database is determined by its support. The support is calculated by ratio of frequency and database size. The pattern is called frequent if its support is greater or equal to minimum support threshold value. The basic frequent pattern mining algorithm is apriori. The apriori algorithm generates candidate sets and use the minimum support threshold value to find the interesting frequent pattern .Here the term interesting may differ from user to user. Some frequent pattern may be important for specific user while the same pattern is not useful to other user.

2. DIFFERENT ALGORITHM FOR MINING FREQUENT PATTERNS AND COMPARISON OF ALGORITHMS:

A. APRIORI ALGORITHM

. There are two main steps in apriori algorithm described as below:

1)Join step:
The candidate itemset are generated based on given database. All possible combination of items are created and their frequency is determined from given dataset.

2)prune step:
In this step based on minimum support threshold value the pruning is done. This is based on downward closure property which state every subset of frequent itemset must also be frequent.

Here number of candidate sets are generated which requires rapid scan on database and storage space. Here if minimum support is low then many uninteresting patterns are found and if the support threshold value is high then we can miss some interesting patterns.

ADVANTAGES OF APRIORI ALGORITHM::

- Easy and simple to implement

LIMITATION::

- In case of large dataset, this algorithm is not efficient [4].
- Apriori algorithm requires large no of scans of dataset [4].
- Minimum support is provided by user which is uniform or constant through whole transaction.
- Not efficient for large database because it need to scan database for several times to generate candidate set which is time consuming process.

IMPROVEMENT IN APRIORI::

- Partitioning: for any itemset i.e. frequent in database, then that itemset must be frequent in at least one of the partition of database [6].
- Transaction Reduction: transactions that do not consist of frequent item sets are of no importance in the next scans for searching frequent item sets [14].
- We can reduce number of scan based on hash function using hash table.

B. HASH BASED FREQUENT ITEMSET (HBFI):

This technique use hash uses hash table structure which will prune the candidate set of infrequent items in early stage. Thus the database scan will be reduced and we get better efficiency than apriori algorithm. It stores the candidate items in bucket form and calculate bucket count. If the bucket count is less than minimum support then it will remove that bucket.

An efficient hash based algorithm for the candidate generation. The number of candidate 2-itemsets generated , in orders of magnitude, smaller than that by apriori method, thus resolving the performance bottleneck of apriori. This algorithm scans the database once utilizing an enhanced version of priori algorithm. The generation of smaller candidate sets can let the transaction database size to be trimmed, thereby reducing the computational cost for later iterations significantly. Hashing to achieve maximum performance, with respect to both execution time and memory usage. Hash based apriori implementation used a data structure that directly represents a hash table, that proposes overcoming some of the weaknesses of apriori algorithm by reducing the number of candidate k-item sets. Applying Hashing data structure hence the execution time is reduced .Thus the performance of Apriori algorithm with hashing is improved with respect to execution time and memory size.

Hash function is:

$$h(k) = ((\text{order of item X}) * 10 + \text{order of item Y}) \bmod n \quad [2]$$

Where n=2m+1 ,M=table size (1)

Figure 2.1: hash based frequent itemset mining algorithm[3]

Hash based Frequent Itemset Mining Algorithm:

The algorithm starts with a candidate itemset of one.
C_k: Candidate item set of size k
L_1<- frequent 1-itemsets
Generate candidate for every L_k do begin
C_{k+1} <- candidate(L_k) #New candidates
Join Step: C_k is generated by joining L_{k-1} with itself
For all transactions t $\in D$ do begin
C_k <- subset [C_k, t] #Candidates contained in t
For all candidates c $\in C_t$ do
c. count++;
Prune Step: Any (k-1) –Subset of infrequent itemset must be infrequent.
L_k {c $\in$C | c.count>= minsup}
Final $L_k$as frequent item from dataset.

ADVANTAGES
:

- Reduced the size of candidate k-item sets.
- Reduced the number of scans on the database.
- Cutting off the large candidates which cause high I/O cost

LIMITATIONS:

- Hash collision may occur which will result in primary clustering. hash collision means the single hash key assigned two or more values.

There are two techniques to remove hash collision:

1. Chaining
2. open addressing:

Which include 3-types as below:

2.1 *Linear probing:*

*T*he interval of probe is fixed

2.2*Quadretic probing:*

Interval between probe are increased using addition of successive output.in this quadratic polynomial is used.

2.3*Double hashing:*

Interval between probe are decided using hash function

Example of this algorithm:

Table 2.1:Transaction Database Table[10]

TID	List of Transactions
T100	I1,I2,I5
T200	I2,I4
T300	I2,I3
T400	I1,I2,I4
T500	I1,I3
T600	I2,I3
T700	I1,I3
T800	I1,I2,I3,I5
T900	I1,I2,I3

Now, the further assignment of item sets to the buckets will be done with using of Hash Function:

h(x, y) = ((order of x) * 10 + (order of y)) mod 7

Table 2.2:Hash Table For Candidate2 generation item sets[10]

Bucket	0	1	2	3	4	5	6
Bucket count	2	2	4	2	2	4	4
Content	{I1,I4}	{I1,I5}	{I2,I3}	{I2,I4}	{I2,I5}	{I1,I2}	{I1,I3}
	{I3,I5}	{I1,I5}	{I2,I3}	{I2,I4}	{I2,I5}	{I1,I2}	{I1,I3}
			{I2,I3}			{I1,I2}	{I1,I3}
			{I2,I3}			{I1,I2}	{I1,I3}

C. DOUBLE HASH BASED FREQUENT ITEMSET MINING (DHBFI):

Double hashing function uses two hash functions to generate frequent item set. In direct hashing the hash collision may occur. This limitation is recovered in double hashing technique. The second hash function is called probing function which remove collision.

First apply hash function to allocate buckets then if collision occur use second hash function.
For ith probe value having k collision can be remover by function.

$$h(k,i)=(h(k)+i*2)\bmod m \text{ [2]} \qquad (2)$$

D. AN ALGORITHM BASED ON BOOLEAN MATRIX (ABBM):

Discovering frequent item sets is the key process in association rule mining and traditional association rule algorithms adopt an iterative method to discovery, which requires very large calculations and a complicated transaction process. A new association rule algorithm called ABBM is proposed .which adopts a Boolean vector "relational calculus" method to discovering frequent item sets. Experimental results show that this algorithm can quickly discover frequent item sets and effectively mine potential association rules from data. Main features of this algorithm are that it only scans the transaction database once, it does not produce candidate item sets, In addition, it stores all transaction data in bits, so it needs less memory space and can be applied to mining large databases to find association rules from it.

E. comparison of different frequent pattern mining technique:

Table 2.3:Different frequent pattern mining techniques

Different algorithms	Used Data structure	Technique	Advantages	Limitations
APRIORI	Array	Use candidate generation with join and prune steps	Very easy and basic concept for frequent pattern mining. useful for sparse and dense data structures	Space and time complexity is high and it require more number of database scan to generate candidates
FP-GROWTH	Tree	Using minimum support it construct frequent dataset tree	2database scan are required. Useful in medium database	If large database the FP tree structure become complex and require complex data structure
SAMPLING	Array	Consider lower threshold value and validate whole database by using small sample of it	Less memory and less time required	Do not give accurate result
PARTITIONING	Array	Based on local frequent itemset and partition the database for this purpose	Useful for large database because it partitions complex data	Time complexity is high because it finds local frequent data and then global frequent data
DHP	Array	Use hashing technique	Useful in small database and medium database and better speed then apriori algorithm	Hash collision may occur
DHBFI	Array	Use two hashing functions	Better than direct hash algorithm to avoid collision	Secondary clustering occur because of quadratic probing
ABBM	Array	Use bit stream to save data	Better than apriori algorithm	Less efficient than hash based technique
H-MINE	Tree	Use pointer to store data and partition the dataset	Efficient memory utilization. Useful for dense data.	Time complexity is greater than other algorithms.

3.CONCLUSION

Hash function based algorithm is better than apriori algorithm .But primary clustering and hash collision may occur in hash based algorithm. So new algorithm using two hash functions generated. Using quadratic probing function as second hash function will create secondary clustering problem.so double hashing technique removes both primary and secondary clustering. New method is required to improve efficiency related to insertion,deletion and search of double hashing technique.

REFERENCES

[1] Ms Shweta, Dr. Kanwal Garg "Mining Efficient Association Rules Through Apriori Algorithm Using Attributes and Comparative Analysis of Various Association Rule Algorithms" ijacsse ISSN: 2277 128X Volume 3, Issue 6, June 2013 page:306-312

[2] L.Padmavath,V.Umarani "An Efficient Association Rule Mining Using the H-BIT Array Hashing Algorithm "International Journal of Advanced Research in Computer Science and Software Engineering ISSN: 2277 128X Volume 3, Issue 1, January 2013 pages:410-419

[3] P. Asha, Dr. T. Jebarajan, Varun Cyriac Thomas "Efficient mining of Frequent Item sets and Association Rules" Australian Journal of Basic and Applied Sciences May 2014, Pages: 510-517

[4] Mamta Dhanda, Sonali Guglani, "Mining Efficient Association rules Through Apriori ", In: Proceeding of IJCST, ISSN 0876-8491, Vol. 2, Issue 3, September.

[5] J.Suresh,P.Rushyanth,Ch.Trinath "Generating associations rule mining using Apriori and FP Growth Algorithms " International Journal of Computer Trends and Technology (IJCTT) - volume4Issue4 –April 2013 pages :887-891

[6] Jogi.Suresh, T.Ramanjaneyulu, "Mining Frequent Item sets Using Apriori Algorithm", International Journal of Computer Trends and Technology, ISSN 2231-2803, Vol. 4,Issue 4, April 2013.

[7] Rachna Somkunwar, "A Study on Various Data Mining Approaches of Association Rules" International Journal of Advanced Research in Computer science and Software Engineering, ISSN 2277-128X, Volume-2, Issue-9, September-2012. Page-141-144

[8] K.Vanitha and R.Santhi "using hash based apriori algorithm to reduce the candidate 2- item sets for mining association rule" Journal of Global Research in Computer Science Volume 2, April 2011pages :78-80

[9] Hanbing Liu and Baisheng Wang "an association rule mining algorithm based on a Boolean matrix" Data Science Journal, Volume 6,9 September 2007 pages :559-669

[10] Jiaweihan,Michelinekamber, Jianpei "Data mining concepts and techniques" morgan kaufmann publishers in an imprint of Elsevier third edition