ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 20, April 2016

Comparison Of Apriori And Partition Algorithms In Extracting Frequent Items

M.Subithra[1],
M.Phil Research Scholar,
Department of Computer Science &Engg,
Alagappa University, Karaikudi, India.
[1]subithra30@gmail.com

Dr.S.S.Dhenakaran[2],
Professor,
Department of Computer Science &Engg,
Alagappa University, Karaikudi, India.
[2]ssdarvind@yahoo@gmail.com

***Abstract-* Now a days, massive amount of information flows in the internet, Especially about E-commerce. Most of people purchase the product from the room to anywhere in the world. Economic research analyzes the behavior of people for increasing the sales and manufacturing. Data mining technique is one of the familiar concepts for gathering and discovery the knowledge from the large amount of dataset. For increasing the sales, the analyzer applies various techniques for finding the frequent item set. Apriori is one of the most common techniques for finding the frequent item set. This algorithm is used to gather the data for frequent data usage or data flow of the domain. The large amount of data split into different sets that the process is called partitioning algorithm. In this paper, the numerical dataset is applied in the Apriori as well as partition algorithm and justify the performance of discovering the frequent item set.**

***Keyword-*: Aprior Algorithm, Frequent item set, Partitioning Algorithm.**

I.INTRODUCTION

Most of companies, organizations and government sectors are processing their information with help of software, massive amount of information stores into database. The database stores the historical information as well as current affairs. Generally, the knowledge discover from the large amount of information with the help of mining techniques that is called data mining techniques. The data mining techniques assist to discover the knowledge from the collection of large amount data.

The data are stored in to centralized database and acts as a large repository that is called data warehouse. The knowledge discovers from the data warehouse through data mining techniques [4][8]. There are several data mining techniques such as Association, Classification, Clustering, Prediction, Sequential patterns and decision making; they are applying into dataset to dick the knowledge.

Association is one the familiar task to discover patterns and generate the association rule[1]. In the association mining, frequency item set is the basic operation for generating the association rule[5]. Researchers Contribute various methods[6][7][8] to find the frequent item set.

Apriori is one of the familiar algorithms for finding the frequent item set. This algorithm uses the bottom up approach[13],where frequent subsets extended one item at a time. It is design to operate on database containing transaction. It has some limitations while applying into large volume of dataset.

PAFI (Partition Algorithm Frequent Items) overcomes the limitation of Apriori algorithm. This is to do the partition the database into transaction in matrix format and find the frequent item set. The proposed work analyzes the performance of Apriori and PAFI algorithm[13]. The analyze result shows the performance of PAFI and it is better than apriori

II.APRIORI ALGORITHM

The main objective of Apriori algorithm is to extract the frequent item set in the sets of transaction from the large dataset. Mainly , the process of finding frequent item set has to be followed the two condition. One is to satisfy the minimum support value and confidence value. The minimum support value decides the lower bound of the frequency item set.

The confidence value measure the certainty of the frequent item set. The confident value is used to generate the association rules. The sets of item has generated with minimum support value. Any subset of frequent item set must be frequent item set. The Apriori algorithm carryout a breadth-first search in the seek space by generating candidate k+1-itemsets from frequent k item sets. The occurrence of frequent

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 20, April 2016

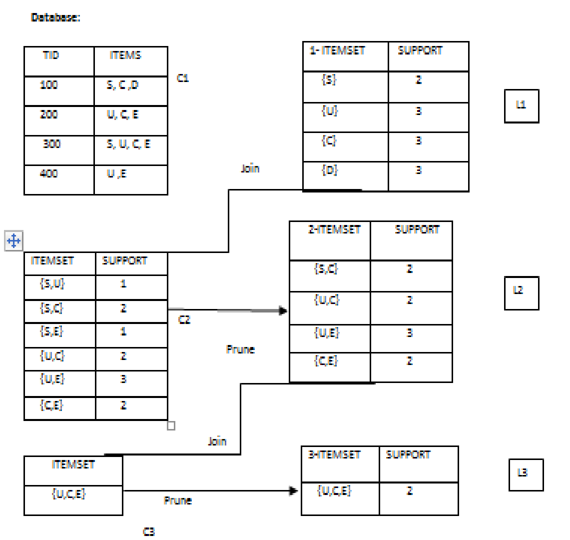

Fig 1. Searching frequent item set mining.

The occurrence of frequent items in each transaction counts the frequent item set.

Pseudocode of Apriori algorithm:

Pass 1

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

Pass *k*

1. Generate the candidate itemsets in C_k from the frequent itemsets in L_{k-1}
    1. Join L_{k-1} *p* with L_{k-1}q, as follows: insert into C_k select $p.item_1$, $p.item_2$, . . . , $p.item_{k-1}$, $q.item_{k-1}$ from L_{k-1} *p*, L_{k-1}q where $p.item_1 = q.item_1$, . . . $p.item_{k-2} = q.item_{k-2}$, $p.item_{k-1} < q.item_{k-1}$
    2. Generate all (*k*-1)-subsets from the candidate itemsets in C_k
    3. 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}
2. Scan the transaction database to determine the support for each candidate itemset in C_k
3. Save the frequent itemsets in L_k

III. PARTITION ALGORITHM

The partitioning algorithm makes the overlapping partitions, D1,D2,D3….Dn from the splitted transactional dataset[11] T, The algorithms decrease the number of dataset inspects to two. During the first inspection, the algorithm discovers all item sets in each partition. Discovered local frequent item sets are consolidated into the global candidate item sets. During the second inspection, these global item sets are counted to verify if they are large across the entire dataset. The partitioning algorithm gets better in the performance of discovering frequent item sets and also having several advantages. Small partitions fits into main memory than that large partition. The algorithm decrease the size of the candidate item set due to reduction of size of the candidate items sets. In addition, the algorithms entail only two inspections on the dataset.

However, the partition algorithms increases the size of There are many types of clustering methods presented, and each of them may form a different grouping of a dataset. The selection of a certain method will corresponds to the type of output preferred, the performance of the algorithm based on the system resources obtainable for size of the data. In general, clustering methods[2] may be splitted into two categories based on the cluster formation which they create. The non-hierarchical method splits a dataset of N objects into M clusters, with or without overlap. These methods are occasionally spitted into partitioning methods, in which the classes are mutually exclusive, and the less common bunching method, in which overlap is permitted.

Each object is a member of the cluster with which it is most related; however the threshold of resemblance has to be defined. The hierarchical methods creates a set of nested clusters in which each pair of objects or clusters is gradually nested in a larger cluster up to only one cluster remains. The hierarchical methods can be further splitted into agglomerative or divisive methods. In agglomerative methods, the hierarchy is constructed in a series of N-1 agglomerations, or Fusion, of pairs of objects, beginning with the un-clustered dataset.

The less common divisive methods start with all objects in a single cluster and at each of N- 1 steps splits some clusters into two smaller clusters, until each object exist in its own cluster.

The partitioning methods generally result in a set of M clusters, each object belongs to one cluster. Each cluster may be refereed by a centroid or a cluster representative; this is some sort of summary depiction of all the objects presented in a cluster. The precise structure of this

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 20, April 2016

description based on the type of the object which is being clustered.

In case where real-valued data is available, the arithmetic mean of the attribute vectors for all objects within a cluster provides an suitable representative; alternative types of centroid may be required in other cases, e.g., a cluster of documents can be represented by a list of those keywords that occur in some minimum number of documents within a cluster. If the count value of the clusters is huge, the centroids can be further clustered to creates hierarchy within a dataset.

The table1 shows the number of transaction and item set.

TID	ITEMS
T1	S,U,D
T2	U,D
T3	U,C
T4	S,U,E
T5	S,C
T6	U,C
T7	S,C
T8	S,U,C,D
T9	S,U,C

For a given set of transactions in the database D, it applies partition algorithm in order to find clusters based on the number of transactions. Here we are getting 2 clusters CL1 and CL2.

After creating cluster using PAFI algorithm, now implement the transaction reduction algorithm to every cluster i.e. CL1 and CL2 but here CL2 has few number of transactions that is less than the threshold value so we are deleting the transactions in CL2 and submit transaction reduction algorithm only on the transactions in CL1. As shown in Figure 2, the affair cluster i.e. CL1 has 7 affairs, CL1={T1,T3,T5,T6,T7,T8 ,T9}, the item sets is I={S,U,C,E} and the minimum support is 2.

As shown in Figure 2, the affair cluster i.e. CL1 has 7 affairs, CL1={T1,T3,T5,T6,T7,T8 ,T9}, the item sets is I={S,U,C,E} and the minimum support is 2.

CL 1	
TID	ITEMS
T1	S,U,D
T3	U,C
T5	S,C
T6	U,C
T7	S,C
T8	S,U,C,D
T9	S,U,C

CL2	
TID	ITEMS
T4	S,U,E
T2	U,D

T ID	ITEMS
T1	S,U,D
T3	U,C
T5	S,C
T6	U,C
T7	S,C
T8	S,U,C,D
T9	S,U,C

Fig 2. Set of transactions in the database with partition

Pseudo code of partition algorithm:

Input:D=dataset

K= the number of centers

C=initial centroids

Output: Set of k representing a good partitioning of D

database and produce the frequent pattern.

1: Select the initial data set

2: for all data point di € D do

3: assigned Center =di. center

4: assigned Partiton= di. partition

5: for all center ci € C do

6: apply on the Item set Ii €I

7: Xn ,n=1,2,3......N

8: A partition P of an interval I is a set of M blocks,

9: P(I)={Bm ,m € M },M={1,2...M}

10: where the blocks are sets of data cells defined by

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 20, April 2016

index sets Nm:: Bm ={Xn,n € Nm}

11: enter key (n) for partition.

12: Find count (item set)

13: if(count (item set) is Even())

{If key (even)

{Partition in n/2 sets

}Else

{Partition in n+/2 sets

}14: if(count(item set) is Odd())

{If key(even)

{Partition in n+/2 sets

}Else

{

Partition in n /2 sets}

15: find frequent pattern for the local

partition until all local partition finishes

16: Combine all local partition and find the

global partition.

17: Finish

IV.LIMITATIONS OF APRIORI

In Apriori, Time complexity is more in case of finding frequent itemset from the large amount of dataset. It does several investigation over the dataset to generate frequenct itemset. For frequent itemset generation process, it utilizes the more memory and Input/Output cost increased.

V.PERFORMANCE ANALYSIS

In this paper, the entire database is splitted into partitions of various sizes, each partition may be called as a cluster. Each cluster is considered one at a time by loading the first cluster into memory and calculating large item sets and the consequent support counts. Then the second cluster is considered as same process followed in the first cluster and the cumulative support count is calculated for the cumulative large item sets.

This process is continued for the entire set of clusters and finally we have the whole large item sets and the corresponding cumulative support counts. This approach reduces buffer memory requirement since it considers only a small cluster at a time and hence it is scalable for any large size of the database. For discovering the large item sets it is sufficient to go through the transactions into the clusters.

There is no need to process the entire database again. Hence it decreases the redundant database inspection and increase the efficiency. The repetition of database inspection avoids finding the frequent items. So the time complexity is reduced when compared to Apriori algorithm. So the performance of partition algorithm is good when compare to the Apriori algorithm based on time complexity and consumption of memory space.

VI. CONCLUSION

In association mining techniques, Finding frequent itemset is basic operation for generating the association rule. Researchers have approached various methods to mine the frequent item set. In this paper. Apriori algorithm and PAFI algorithm have taken for analyzing the performance.

The efficiency of PAFI is better than Apriori when compared to iteration process, time complexity and utilization of memory space.

Reference

[1] Agrawal R, Imielinski T, Swami A, “Mining association rules between sets of items in large databases”. In: Proc. of the l993ACM on Management of Data, Washington, D.C, May 1993. 207-216

[2] D.Kerana Hanirex, Dr.M.A.Dorai Rangaswamy:” Efficient algorithm for mining frequent item sets using clustering techniques.” In International Journal on Computer Science and Engineering Vol. 3 No. 3 Mar 2011. 1028-1032

[3] Margatet H. Dunham. Data Mining, Introductory and Advanced Topics: Upper Saddle River, New Jersey: Pearson Education Inc.,2003.

[4] Tong Qiang, Zhou Yuanchun, Wu Kaichao, Yan Baoping, “ A quantitative association rules mining algorithm”. Computer engineering. 2007, 33(10):34-35

[5] Wael A. AlZoubi, Azuraliza Abu Bakar, Khairuddin Omar,” Scalable and Efficient Method for Mining Association Rules”, International Conference on Electrical Engineering and Informatics 2009.

[6] Wael Ahmad AlZoubi, Khairuddin Omar, zuraliza Abu Bakar “An Efficient Mining of Transactional Data Using Graph-based Technique”,3rd Conference on Data

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 20, April 2016

Mining and Optimization (DMO) 2011, Selangor, Malay

[7] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules," in Proceedings of the 20th VLDB Conference, 1994, pp.487–499.

[8] Arun K Pujari. Data Mining Techniques (Edition 5):Hyderabad, India: Universities Press (India) Private Limited, 2003.

[9] Margatet H. Dunham. Data Mining, Introductory and Advanced Topics: Upper Saddle River, New Jersey: Pearson Education Inc., 2003.

[10] Jiawei Han. Data Mining, concepts and Techniques: San Francisco, CA: Morgan Kaufmann Publishers.,2004.

[11] Akhilesh Tiwari, Rajendra K. Gupta, and Dev Prakash Agrawal "Cluster Based Partition Approach for Mining Frequent Itemsets" InProceedings of the IJCSNS International Journal of computer Science and Network Security, VOL.9 No.6, June 2009

[12] R.K. Gupta. Development of Algorithms for New Association Rule Mining System, Ph.D. Thesis, Submitted to ABV-Indian Institute of information Technology & Management, Gwalior, India, 2004

[13] Son N. Nguyen, Maria E. Orlowska,A Further Study in the Data Partitioning Approach for Frequent Itemsets Mining, 17th Australasian Database Conference (ADC 2006),