International Journal of Research and Review
www.ijrrjournal.com E-ISSN: 2349-9788; P-ISSN: 2454-2237

Research Paper

Apriori Algorithm Using Map Reduce

Prof. Mrs. Sandhya S.Waghere[1], Sanchita Sonar[2], Shweta Kawad[2], Karishma Murudkar[2]

[1]Assistant Professor, Information Technology, Pimpri Chinchwad College of Engineering, Pune, India,
[2]BE Information Technology, Pimpri Chinchwad College of Engineering, Pune, India,

Corresponding Author: Shweta Kawad

ABSTRACT

Data mining is mainly used to extract the important information from large databases. On the other hand cloud computing provides the large data storage but now days, data is increasing day by day. To handle this much data and learn patterns from it has become a challenging part. Because, to learn the patterns or knowledge from these bulk amount of datasets available systems take more time and require more resources. Hence to improve the speed and reduce required cost in order to improve systems efficiency we are going to provide an algorithm called FIM which includes the map- reduce programming for frequent itemset mining. We are implementing three successive map reduce to find association rules.

Key concepts: Apriori, map reduce, association rule mining, frequent itemsets.

I. INTRODUCTON

Now a day, finding out useful knowledge from the huge data is popular topic in recent years. Frequent itemset mining is process of finding the regularities between items. Basically it finds the set of items which frequently come together. It is mainly used in market basket analysis. It helps to take business decisions like which items should be selected for seal, what should be coupon policy and how to arrange the items on self in order to maximize the selling rate. During previous days several organizations used to store their data on databases. But, databases are not that much compatible or it doesn't provide the functionality that provides the exact related information that user wants. So to overcome this problem frequent itemset mining by considering the association rules comes in to existence.

Available methodologies for frequent itemsets mining does not provide the high computational rate due to large amount of input and output data and this data is increasing day by day and so that Frequent Itemset Mining has become an important issue in sequence mining and association rule mining. To overcome this problem we are going to provide an algorithm which uses the Map Reduce programming to provide the load balancing and make system more efficient. Here we used three subsequent map reduces to implement this algorithms. Map reduce methodology provides features like automatic parallelization, load balancing and data distribution. It takes less time for computation. Therefore, it increases the efficiency of system.

II. METHODOLOGY

In this system, three Map Reduce jobs are implemented to complete the mining task. Consider a database containing transactions stored in files. All the transactions of database are divided and stored in the input files by HDFS file system

across various node in Hadoop cluster. This input files are input for the first Map Reduce job. Mappers reads the transactions from its local input file and gives the item count. The first Map Reduce job find out all frequent items or frequent one-itemsets. In this phase, the input of Map tasks is an input file and the output of Reduce tasks is all frequent one-itemsets. So we get the key value pairs as an output where item is key and count of each item in a database is value of corresponding items. We calculate some threshold value called support for each items by using output of first step. The second Map Reduce job again scans the database to generate k-itemsets by pruning infrequent items in each transaction by using output of first Map Reduce. By making combinations of items we get count of itemsets where all items from itemset occur in transactions. In Third map reduce we find out the most frequent k-itemsets using input from the output of second map reduce. We find the frequency of each item in frequent itemsets.

MAP-REDUCE 1:
The first MapReduce job is responsible for creating all frequent one-itemsets.
Input-
Each mapper sequentially reads each transaction from its local input split, where each transaction is stored in the format of pair<LongWritable offset, Text record>.
Output-
Global frequent one-item sets are generated and written in the form of pair
<Text item, LongWritable count>.

MAP-REDUCE 2:
The second job marks an itemset as a k-itemset if it contains k frequent items ($2 \leq k \leq M$, where M is the maximal value of k in the pruned transactions).
Input-
Output of first map reduce is given as input to second map reduce.
<Text item, LongWritable count>
Output-
<IntWritable itemNumber, MapWritable<ArrayWritable k-item, LongWritable Sum>>.

MAP-REDUCE 3:
The third map reduce find out the association rules using k-frequent itemsets. This rules are nothing but the dependency of items in transactions. We find those rules by analysis the huge database which helps to improve business logic.

Example
Let's find the association rules using this frequent itemsets mining algorithm. Consider a sample transaction database for understanding the working of FIM algorithm.
Consider following database to find FIM.

ID	ITEMSET
1	M O N K E Y
2	D O N K E Y
3	M U C K Y
4	M A K E
5	C O O K E

For this transaction lists first map- reduce performs mapper and reducer function and calculate the frequency count for each item.
Now first step is over. This output is stored in temporary files which will be used for further calculations. Second map reduce again scans the whole transactions and calculate frequent k-itemsets with count.

ITEMSET	COUNT
M	3
O	4
N	2
K	3
E	4
Y	3
D	1
A	1
U	1
C	2

As we have considered number of transactions is 5.
Let’s assume minimum support value which is given input is 60%

Support = minimum support * number of transactions

= 60/100 * 5

= 3

Support =3

Using this support value prune all itemsets which has lesser frequency count than support value.

Pruned output will be like shown in below table.

ITEMSET	COUNT
M	3
O	4
K	3
E	4
Y	3

Similar way we will find k-itemsets by pruning infrequent itemsets from the database.

We will get following itemsets for 2 size of itemset.

ITEMSET	COUNT
M, K	3
O, K	3
O, E	3
K, E	4
K, Y	3

For size 3 of itemset:

ITEMSET	COUNT
O, K, E	3

Now create association rule using support and confidence.

O^K => E

O^E => K

K^E => O

E => O^K

K=> O^E

O=> K^E

Compare all this association rules with minimum confidence value

As we have considered 80% minimum confidence value, we will get following final association rules.

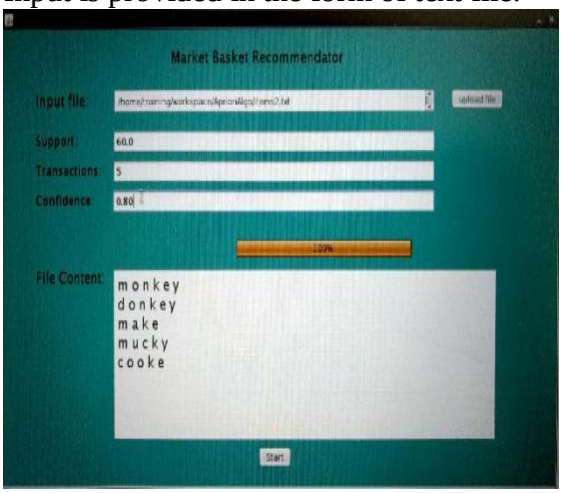
O^K => E

O^E => K

III. RESULTS

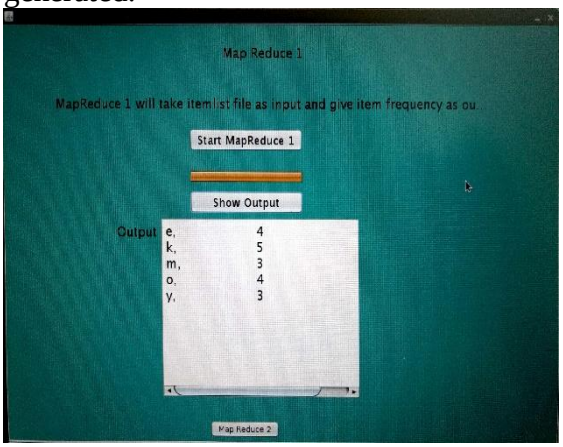
Main Frame

Input is provided in the form of text file.



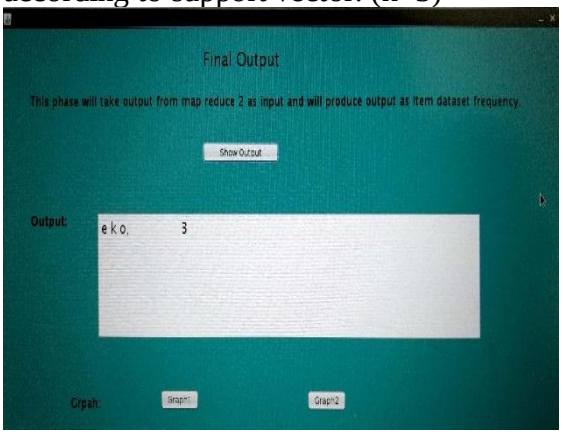
Output of 1st Map Reduce

Frequency count of one itemset is generated.

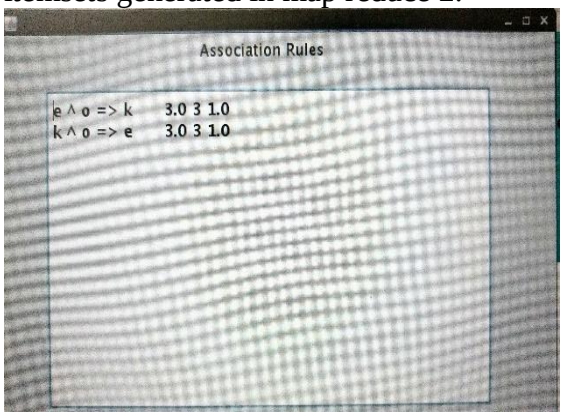


Output of 2nd Map Reduce

K itemset are generated and pruned according to support vector. (k=3)



Outputs of 3rd Map Reduce
Association Rules are generated using the k itemsets generated in map reduce 2.



IV. CONCLUSION AND FUTURE SCOPE

We have implemented frequent itemset mining (FIM) algorithm using map reduce paradigm. Here we used three successive map-reduce. First map-reduce find out one-itemset and its frequency count. Second map-reduce find out k-itemsets pruning infrequent itemsets by using output of first map-reduce. This increases the efficiency of our algorithm to find association rules. It helps to prevent unnecessary combination of itemsets which occurs lesser time than threshold value. This is how we tried to optimize our algorithm by applying three successive map-reduce. In third map-reduce we calculated association rules using confidence value. As we have used map reduce paradigm which provides features like automatic parallelization, load balancing and data distribution. Future scope of this project is very vast it can be effectively used in various sectors like in health sector. This algorithm can be used in adverse drug reaction detection. It is also used in Oracle bone Inscription Explication which is an oldest writing in the world

V. ACKNOWLEDGEMENT

Our sincere gratitude to Mrs. Sandhya Waghere, Assistant Professor in Department of Information Technology for her valuable support and guidance.

VI. REFERENCES

- Yaling Xun, Jifu Zhang, and Xiao Qin, "FiDoop: Parallel Mining of Frequent Itemsets Using MapReduce" in IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2016
- Sandhya S Waghere, Pothuraju Rajarajeswari, "Parallel Frequent Dataset Mining and Feature Subset Selection for High Dimensional Data on Hadoop using Map-Reduce" in International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 18 (2017) pp. 7783-7789.
- Sandhya S. Waghere, Pothuraju Rajarajeswari, "A Survey on Achieving Best Knowledge from Frequent Item set Mining using Fidoop" in International Journal of Computer Applications (0975 – 8887) Volume 171 – No. 9, August 2017.
- lugendra Dongre, Gend Lal Prajapati, S. V. Tokekar, "The Role of Apriori Algorithm for Finding the Association Rules in Data Mining" in 2014 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT)
- Angelina A. Tzacheva, midhun M. Sunny and Pranava Mummoju, "MR-Apriori Count Distribution Algorithm for Parallel Action Rules Discovery" in 2016 IEEE International Conference on Knowledge Engineering and Applications
- Sandhya Harikumar, Divya Usha Dilipkumar, "Apriori Algorithm for Association Rule Mining in High Dimensional Data" in 2016 IEEE International Conference on Data Science and Engineering (ICDSE)

How to cite this article: Waghere SS, Sonar S, Kawad S et al. Apriori algorithm using map reduce. International Journal of Research and Review. 2018; 5(5):129-132.

******