International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-8 Issue-4, February 2019

Efficient Algorithm using Big Data for Frequent Itemsets Mining

Srinivasa Rao Divvela, V Sucharitha

Abstract: ***Future trends are being estimated with the help of tools in data mining which allows the making of decisions to be data driven and analyze them carefully with the corresponding tools. In various fields of mining the most important practice of mining of data is the Associate-rule mining. Major issue in any of the techniques being the generation of the frequent data-item sets which has to be solved efficiently. Many techniques have been put forth for this only purpose of itemset generation like Apriori-algorithm, FP_Growth-algorithm, and many other solutions are being offered to solve the issue. Many outsets of the problem yet to be fully implemented such as large clusters solving and distribution along with parallelization (automatic) etc. Many of these issues can be solved with the implementation of Framework of MapReduce on Improved Apriori algorithm. Lessening of time due to parallel executions can be achieved with the help of this. This procedure considerably decreases the time of execution and also a significant rise in efficiency is observed.***

Keywords: MapReduce, Improved Apriori, mining, Frequent data-item sets.

I.INTRODUCTION

Huge information is created from the different fields of innovations, for example, IT enterprises, Health care ventures, government organizations, divine information and a lot more.The Past frameworks ofFidoop are adequate to discover and visit theitemsets which are present in parallel utilizing Hadoop_mapreduce along with the F.I.U.T mining calculations. By the utilization of regular ultarmetric trees (F .I .U .T) it is conceivable to accomplish packed capacity in comparision to the utilization of conventional FP trees. The major setback of the existing framework is, for given expansive datasets the strategy for dividing of information is experience the ill effects of I/O and mining endeavors because of intemperate exchanges transmitted between hubs. Second, FIUT calculation gives information spillage issue and decreases execution. The existing systems cause quite a pickle for the users as they arenot time efficient and space efficient they always arise time complexity and space complexity issues which has to be handled by the user himself.

II.PROPOSED FRAMEWORK/SYSTEM

1. Problem_Defination

Analysis of all the issues within the existing framework meanwhile hunting different issues identified with information dividing and stack adjusting.

Manuscript published on 28 February 2019.
***Correspondence Author(s)
Divvela Srinivasa Rao, Sr. Assistant Professor, Laki Reddy Balireddy College of Engineering, Mylavaram, A.P. India
Dr. V. Sucharita, Professor, Department of Computer Science and Engineering, Narayana Engineering College, Gudur, A.P. India

Examination of productive approaches to discover visit itemsets with decreased calculation time through the implementation of parallelism, stack adjusting.

2. System Architecture

The primary assignments oversee by employment administrator are information apportioning and grouping, plans the undertakings. Name nodes areexperts who oversee document frameworks.

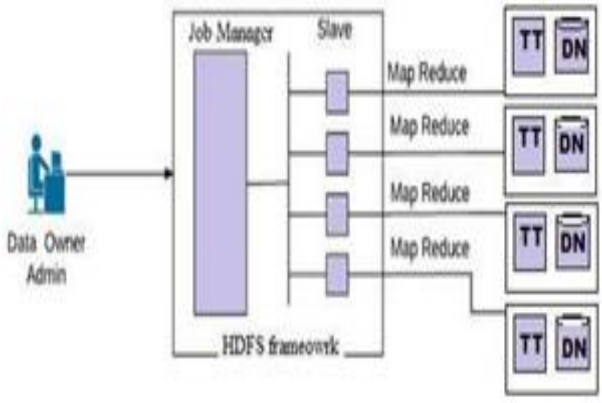

Fig 1: System_Architecture

III.IMPLEMENTATION

1. Proposed framework working

To achieve and anable the concept of parallelism, the visit item-set mining execution is done with MapReduce method of programming model. Subsequently, downsides of conventional strategies can be tended to and effective information circulation, computerized parallelization, stack adjusting can be accomplished. Thus proposed framework is executed with improved and advancedApriori calculations by the usage foMapRdecue. The productivity as far as time and memory can be accomplished utilizing this methodology.

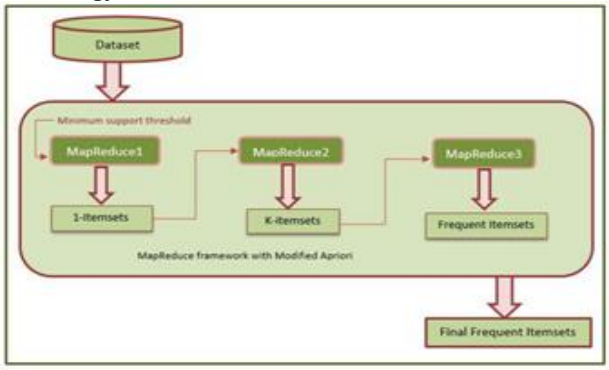

Fig 2. System_WorkFlow

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

1. Algorithm for Frequent1 itemsets.

- Input given will be a particularDataset DB along with a minimum support.
- Output will be given as 1-itemsets.

Mapper_Algorithm:

Function Mapper (Key offset, value Db) //Tr is the Transaction in Db

Step 1: For all Tr, split each Tr and assign it to the items.

Step 2: Output (item, 1); Reducer

Algorithm:

Function Reducer (Key K^{th}-Itemset, Value 1)

Step 1: Let us consider sum=0.

//Sum is the minimum Support for this itemset.

Step 2: For each K^{th}-itemset do sum+=1.

Step 3: Output (1-itemset, sum); //item as 1-itemset.

Step 4: If sum is greater than or equal to the minimum support then add items to F-list. This list contains the frequent 1 itemset and their count

The present algorithm 1 is used for the finding of the 1_itemsets which are frequent in nature in the dataset. This is useful as it is given as input for the next algorithms for the processing of the data and progressing of the method which we are implementing.

2. Algorithm for Frequent K itemsets.

Input: Dataset Db, min support.

Output: K-itemsets.

Mapper Algo:

Function Mapper (items, D1)

//Trn is the Transaction in Db

Step 1: For all Trn, split each Trn and assign it to the items.

Step 2: For each item in Items, if the item is not frequent then prune item from Trn.

Step 3: Kth-itemsets will contain all the frequent items in the itemsets.

Reducer Algo:

Function Reducer (Key Kth-Itemset, Value 1)

Step 1: Let us consider sum=0.

//Sum is the min Support for this itemset.

Step 2: For all Kth-itemset do sum+=1.

Step 3: Output (k, Kth-itemset + sum)

The second algorithm utilizes all the 1_itemsets produced by the first algorithm along with the dataset used for the processing of the data and further implementation of the method we are implementing. This in turn gives us the frequent data itemsets as the output which are from the k_itemsets given as input from the first algorithm to the second algorithm. We can simply state that piping procedure takes place in the procedure.

3. Improved Algorithm

Input: Dataset, minsupport

Output: Final k-Frequent itemsets

Step 1: Let k be the variable for select one cluster at a time.

Step 2: Generate items, Number of records N and min support

Step 3: Let f (f1) for finding frequent itemsets

Step 4: f1 =find-req-1itemset (N)

Step 5: use for loop from k=2 to all NStep 6: generate frequent-2 itemsets

Step 7: HT to assign hash table with the maximum size 8.

Step 8: Store candidate itemsets into the HT with Unique values.

Step 9: if (items of HT >= minsupport) then bit vector ==1 else bit vector ==0

Step 10: prune the items with min support

Step 11: x = set-bit-items (HT) and prune the items which is==0

Step 12: modify the items list x that contains 1 from the HT

Step 13: for each Records N in x increment count of all HT

Step 14: fk = items in x with min support

Step 15: end for

IV.RESULTS

Proposed structure execution shows the FIUT (Frequent ItemsetUltrametric Tree) framework use. Separation and Improved Apriori calculation, FIUT sets aside increasingly unmistakable opportunity to execute an occupation. As this figuring concentrating on decay of number of hopefuls sets and decline the measure of yields, fittingly execution time will diminish. The current improved algorithm shows that the time consumption for the advanced algorithm is quite low in comparison with the F .I .U .T algorithm execution. The time is nearly 500-1000 sec variation in the execution for the datasets having data items of value more than 10000 or so. This shows that this method of advanced or improved algorithm is quite ueful foe larger datasets in finding out the frequent data itemsets .

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

Size of Dataset (Number of Records)	Process Timing (Sec)	
	Improved Apriori	*FIUT*
10000	1200	1800
20000	1600	2500
30000	2100	2900
40000	3000	3500
60000	6500	7700

Table I. F .I .U .T and Improved Algorithm time comparison

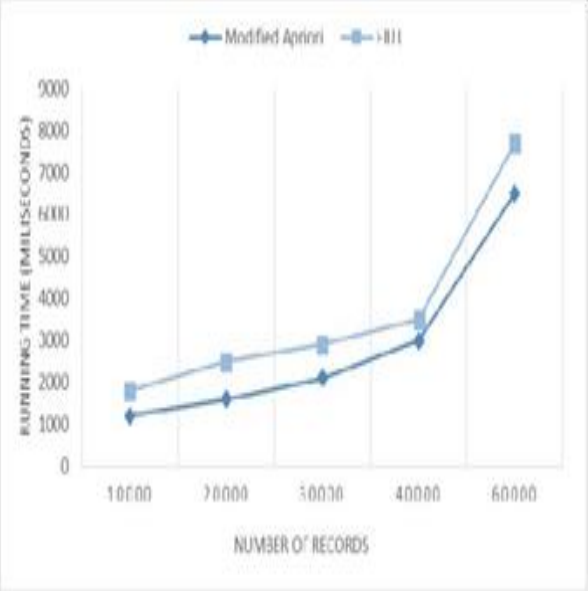

Fig 3: threshold of value 3 (Runtime for various size records with minimum support)

V.CONCLUSION

Investigation of enormous datasets gives forecasts and profitable data that can be utilized for some applications, for example, business choices, feeling examination, showcase bin examination, web logs investigation, tranquilize design(molecular section mining) and in like manner. The new methodology is acquainted with unravel versatility. In this manner Improved calculation demonstrates the rapidly execution of successive itemsets.

REFERENCES

1. YalingXun, Jifu Zhang and Xian Qin, "Fidoop: Parallel mining of frequent Itemsets using MapReduce", IEEE Trans.onsys.man and cybernetics, Vol. 46,No.3, March 2016.
2. Sheelagole and Bharat Tidke, "Frequent Itemset Mining for BigData in social media using ClustBigFIM algorithm", Intl Conf.on Pervasive Computing, IEEE 2015.
3. Marconi K, Lehmann H. Big Data and Health Analytics[M]. BocaRaton:CRC Press, 2014.
4. McKinsey&Company. The big-data revolution in US health care: Accelerating value and innovation [R]. http://www.mckinsey.com/industries/healthcare-systems-and-services/our-insights/the-big-data-revolution-in-us-health-care, 2013.
5. Zahra Farzanyar and Nick Cercone, "Efficient mining of frequent itemsets in social network data based on MapReduce framework", Proceedings of the 2013 IEEE International Conference on Advances inSocial Networks Analysis and Mining.
6. M.-Y. Lin, P.-Y. Lee, and S.-C. Hsueh, "Apriori-based frequent itemset mining algorithms on MapReduce", International Conference onUbiquitous Information Management and Communication, ACM, 2012.

AUTHORS PROFILE

Divvela Srinivasa Rao is currently working as Sr.Assistant Professor in Laki Reddy Balireddy College of Engineering, Mylavaram. He is a Research scholar in KL Universiry. His area of interest is Data Mining and Knowledge Engineering. He has total 13 years of teaching experience Experience in various Engineering colleges in Andhra Pradesh. He has published 10 papers in various national and international Journals, conference proceedings and Conferences.

Dr.V.Sucharita is currently working as Professor in the Department of Computer Science and Engineering in Narayana Engineering College,Gudur. She has obtained her Ph.D degree in the faculty of Computer Science with Data Mining and Neural Networks as specialization from Sri Padmavati Mahila Visvavidyalayam,Tirupathi. She has 17 years of Teaching Experience in various Engineering colleges in Andhra Pradesh. Prior to joining Narayana Engineering College, She worked as Professor in K.L University, Guntur. She has published 60 papers in various national and international Journals, conference proceedings and Conferences. She is an Editorial Board member in reputed journals.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)