International Journal of Research
Available at https://edupediapublications.org/journals
Special Issue on Conference Papers
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 06
March 2018

Review of Apriori Based Algorithms on Mapreduce Framework In Big Data

[1]DR.M.SURESH, [2]SHAIK. JAKEER HUSSAIN, [3]RAJANI PARIMALA
[1]HOD, Dept. of CSE, Mahaveer Institute of Science And Technology, Hyderabad.
[2]Associate Professor, Dept. of CSE, Mahaveer Institute of Science And Technology, Hyderabad.
[3]Assistant Professor, Dept. of CSE, Mahaveer Institute of Science And Technology, Hyderabad.

ABSTRACT— Frequent pattern Mining is an important discovery in data mining tasks.Thus, it has been the subject of numerous studies and research since its concept came . Mostly studies find all the frequent patterns from collection of precise data, in which the items within each datum or transaction are definitely known. But,in many real-life scenario in which the user is interested in only some tiny portions of these frequent patterns. Thus we go for constrained mining , which aims to find only those frequent patterns that are interesting to the user. Moreover, there are also many real-life scenario in which the data are uncertain .In our project, we propose algorithms which will efficiently find frequent patterns and by applying constraint from collections of uncertaindata.

Keywords— MapReduceModel; programming skill for Big Data mining; Big Data analysis; Searching and mining Big Data; Frequent Pattern; Constraints; Uncertain data.

I. INTRODUCTION

A Database involves many different work that help to extract useful knowledge from raw dirty data is known as Knowledge Discovery. The process requires a tough user interaction in order to make client job easy help him to get useful knowledge. This can be done by means of data mining primitives that should include :-

1. The sourcedata
2. The kind of knowledge to bemined
3. Backgroundknowledge
4. Interestingness measures for patternsevaluation
5. The representation of the extractedknowledge

By using a query language useful to apply all above features may result,The implementation is a challenge.Thisgoalisindicatedin[12]whereM anilaintroducesanimportantinteractiveminin gprocess:thatisinductivedatabase which is relational database added with the set of all sentences from a specified class of sentences that are true ofthedata. TheInductiveDatabaseisnaturallyinrule-basedlanguages,suchasdeductivedatabases[1 3,14].Adeductive database is both extensional and in tensional data, thus allowing a higher degree of expressivenessthantraditional relational algebra. This efficiency makes it easy for better representation of domain knowledge and support the steps of the KDD process.

A. *Background*

In the past few years ago data are need to be stored that has increased drastically all over the world. The data generated from web logs, machine logs, human generated data etc. are being stored by companies. This phenomenon is known as "Big Data" and is popular everywhere. This incredibly fast growth of data results in the need to analyse the huge amount of data.And due to lack of proper tools and programs, data remains unused and unutilized with important useful knowledgehidden.

For processing huge amount of data, new tools and approaches have come into role. The most commonly used tool for analysis of Big Data is Hadoop. It is used for batch processing of large datasets. Mining of frequent patterns of itemsets found in large transactional datasets helps in further important data mining tasks like association rule mining, correlations, clustering etc. Apache Mahout is a tool available for mining of frequent patterns from large datasets , a scalable machine

learning library which implements frequent pattern growth algorithm on Hadoop using MapReduce programming model.

B. Motivation

There are many existing data mining algorithms that search interesting patterns from transactional databases of precise data. However, there are scenario in which data are uncertain. Items in each transaction of databases of uncertain data are related to existential probabilities, which shows the probability of these items to be present in the transaction. After comparing with mining from precise data, the search space for mining from uncertain data is much larger due to the presence of the existential probabilities. This problem is increased with the arrival of Big data. In many real-lifeapplications, users may be interested in a tiny portion of this large search space for Big Data mining. Without providing facilities for users to express the interesting patterns to be mined, many existing data mining algorithms return lots patterns out of which only some are interesting.

When the data is uncertain,then each transaction contains items and their existential probabilities[3],[4],[5] .The uncertainty of such expected item can be expressed in terms of existential probability [3],[4]. In uncertain data, each item in a transaction is associated with probability that indicate the possibility that the item exists in the transaction. Normally these types of set of data items is called uncertain dataset. Figure.1 shows an example of transactional dataset of Precise and uncertain data

Thus pruning of unimportant patterns also becomes an important step in this frequent pattern mining.So,we parallelize all mining of frequent itemset to deal with large scale data mining problems. Parallel algorithms are developed to minimize memory use and computational cost on each machine. Recent works of parallelizing Apriori Algorithm suffers from high communication cost between nodes.

To reduce the resource constraints and also restarting computational failures we have used a MapReduce based parallel Apriori Algorithm. This research is used to successfully generate important patterns of itemsets from huge transactional datasets. The patterns that are generated will help in future analysis of other data mining tasks. It aims to yield correct results in minimum considerable of time after processing huge volume of data.

II. LITERATURESURVEY

Author in this paper [2] tried to consider all the existing data mining algorithms for searching information with interesting patterns from transactional databases of precise data.But when the data are uncertain items in each transaction of these probabilistic databases of uncertain data are usually associated with existential probabilities, which shows the possibility of items to be present in the transaction.

In comparision with precise data, the search space for mining data from uncertain data is much larger due to the presence of the existential probabilities. This problem is worsened in Big Data era. And, in some real-life applications, users are onlyinteresteed in little portion of this large search space for Big data mining. So by providing opportunities for users to express their interest for interesting patterns mining.But mostly existing data mining algorithms returns number of patterns out of which only some are interesting. In this paper,we propose an algorithm that (i) allows users to express their interest in terms of constraints and (ii) uses the MapReduce model to mine uncertain Big Data for frequent patterns that satisfy the user-specified constraints. By exploiting constraints properties, our algorithm minimizes the search space for Big data mining of uncertain data, and returns required requested patterns for Big dataanalytics.

In order to use the Map Reduce model several algorithms have been used to mine information from a large space.An important Big data mining and analytics task is Frequent pattern mining which mine the frequently occurring items with consideration of parallel and distributed

computing [8] on large clusters or grids of nodes (i.e.,commodity machines), which consist of a master node and multiple worker nodes. As it shows by its name, MapReduce involves two key functions: "map" and "reduce".One of the probem to uncover hidden knowledge from Big Data is concept drift where statistical properties of the attributes and their target classes shift over time, making the designed trained model less accurate.

SrNo.	Paper title	Author Name	Technique	Advantages	Disadvantages
1	Efficient Mining of Frequent Patterns from	Carson Kai-Sang Leung* Christopher L.	ReverseApriori Algorithm	Handles uncertain data and reduces the generation of candidates	Less efficient in Handling preci se data
2	Result Analysis of Mining Fast Frequent	Shweta Kharat and Neetesh Gupta	Reverse Apriori Algorithm	It works faster than existing Apriori Algorithm & generateless	Still have drawback of more time ,space&
3	Reducing the Search Space for Big Data Mining for Interesting	Carson Kai-Sang Leung* Richard Kyle MacKinnon	Map Reduce Techniqu e	Users only need to focus on map & reduce functions without worrying about partitioning	When you will have OLTP needs, Map Reduce is not suitable
4	Online Association Rule Mining over Fast Data	ErdiÖlmezoğ ulları, Ismail Ari	"Receptor" a complex event processing	They can eliminate unwanted data early in the pipeline,saving	
5	Counterin g the Problem in Big Data Using	Hang Yang, Simon Fong	iOVFDT a Decision tree single tree	It balances accuracy,tree size Speed	Concept drift Issue

A. Algorithm

U- Reverse Apriori

In reverse apriori approach it generate large frequent itemsets which starts by considering a maximum contribution of all the values in pairs.It is obvious to find out these combinations by having a glance at it and generally a minimum support value in thedataset.

The proposed methodology uses reverse Apriori algorithm where we backtrack a database. In order to find maximum number of frequent patterns and step by step will derive the corresponding association rules. In contrary to Apriori, this approach starts with maximum number of collected attributes from database transaction. These collective attributes are compared against the minimum support for the associated rule and is selected and fed into nextstep.

B. Finding Frequent Itemsets Using Reverse- AprioriAlgorithm

This approach is bottom-up because it works entirely opposite to apriori algorithm. In this approach, first find out the pattern by making all possible pairs of itemset and discard the items which does not satisfy the user defined minimum threshold called minimum support minsupp. and evaluate a maximum possible limit of number of items in the dataset thereby generating a huge amount of frequent itemsets satisfying a user specified minimum support.It will slowly and slowly minimise the simultaneously frequent itemset till it gets a set of possible frequent itemsets. Let DS= (A,B,C,D) are the set of items which belongs to the transaction T. The pairs are said to be conjunctive if (A,B) E is user defined support. A pattern P is said to be frequent if minSupp (P) is greater than or equal to a minimum support threshold, denoted as minsupp.On the contrary,

disjunctive patterns are those which contains all the different and irrelevant pairs of sets and therefore should be rejected as they are outliers Disjunctive if (A,B)! E user defined support.For example in generalized terms, let's consider a transaction based on a super market which contains a huge set of items and their occurrence frequency. It has been zeroed in user defined support on milk-made items. Considering a transaction that has all the possibilities of items being paired, Nowthis transaction consists of all the items ranging from–T = {bread, onion, banana, butter, toothpaste, cheese, egg, pasteurized milk, peas,wafers, biscuits .....}

Now user defined support is to bakery products then it's not an intelligent step to take sample combination of all the item sets one by one and then generate candidate-1 item sets and so on. Thus what can be done here is that it will just take only those items which seem to lie in this category of user defined support and that is bakery made products.Thus the conjunctive pattern will contain only those products which fall into this specified range. And the rest of the items are considered as disjunctive patterns since they do not fall under the category of selection and therefore needs to be discarded.

Conjunctive sets= {bread,biscuit,pastries…}Disjunctive sets = {bread, egg, toothpaste, wafers…}.The reverse Apriori is then applied which works faster than the existing Apriori algorithm.

ITEMSET VALUES OF ITEMSET
Temperature Hot, mild, chilling Humidity High, normal, low Pitch Dry, damp
Soccer Yes, no

Here let's assume that John has a fixed user defined support of playing the soccer if and only if the weather is mild and dry. Then by declining the combination of irrelevant and unnecessary items and their values is an effective way to reach onto the decision rather than considering all in all sets. Through these very simple and easy to understand examples, the conjunctive and disjunctive pattern are getting deployed and how they can be diminish the need of higher order candidate generation procedure.

The proposed bottom-up algorithm with conjunctive pattern is: Input:
A database D containing transactions T. Min_support S
Output:
Large frequent item set Algorithm:

1. Scan the database transaction which has some distinct items T = {X, Y, Z, F, P, M, L,S}
2. Find out the conjunctive patterns from thetransaction
3. If X, Y, F €usr-def-sup
   Conj = (X, Y, F)
   Else
   Disj = {P, L, M, Z}
4. Max=con 5. j=0
6. For all further combinations of (max-i) number of attributes
7. Do
8. Generate candidate (max-i) itemsets
9. Frequent (max-i) item sets FPkis generated from candidate (max-i) itemsets
10. Where support count of generated item sets >=min_sup
11. If successful then go tostep13
12. Else j=j+1 and go to step6
13. Return sets of large frequent itemsets
14. End

III. RESULTS ANDDISCUSSION

To calculate the efficiency and effectiveness of the improved algorithm, two algorithms has been performed: Apriori algorithm and Reverse Apriori algorithm. All the experiments were examined on Intel i3 processor, running Microsoft windows 7.The algorithm is implemented in Java1.8.

Below figure show full execution of the projects.

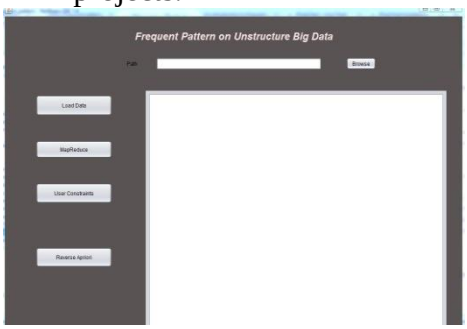


Fig. 1 Home page

Fig.1 show the home page, in home page there are some button, 1st user will click on browse button to select dataset show in fig. 2

Keys	Counts
1	83015
2	20591
3	2110
4	9077
5	33046
6	3279
7	24231
8	84353
9	25391
300	15
301	12
302	37
305	30
306	50
307	34180
308	22
309	19
311	31
315	22
200	745
201	689
202	15
323	17
203	399
204	1502
205	196
206	240
207	407
208	174
329	14

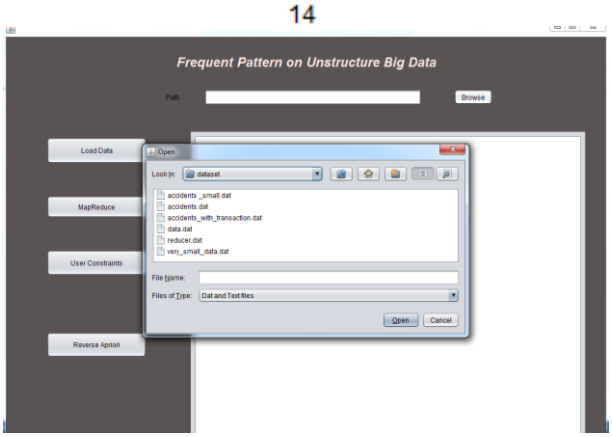


Fig. 2 Selection of dataset

After selecting dataset, user will click on MapReduse phase, output of MapReduse phase shown in fig 5.3.

Fig. 3 OutPut of Map

Reduce After MapReduce user will provide user Constraints that show in fig .4

Time Requirement for 43 Records		
Support %	Apriori	Reverse Apriori
10	300	280
20	320	310
30	333	320
40	350	342
50	400	370

Fig. 4 User constraints

After selection of user constraints, here user select constraints "Type of Road is highway" and output of this constraints are only those data which have attribute "highway". Fig.5 show how many number of transaction are retrieved.

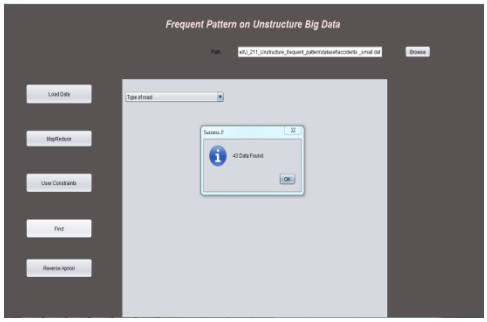


Fig. 5 information Retrieval

Fig.5 show 43 records out of 8 lack are retrieved, now system will calculate frequent pattern from this 43 record, for that purpose we have implemented reverse Apriori algorithm. Output is show in fig.6

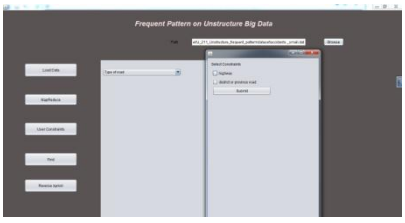


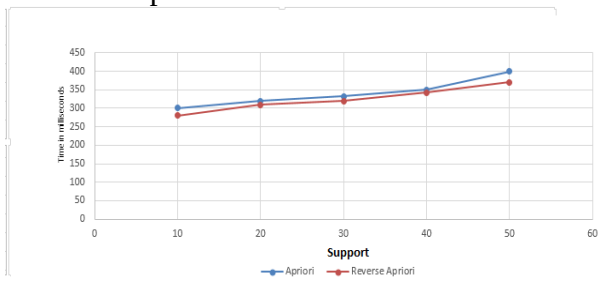
Fig. 6 Frequent pattern output

So the frequent pattern are for "Type of Road is highway" are "Downhill , Monday , Highway" Comparison of the two algorithms on two different measures. Those are mentioned below:

(a) Timerequirement
(b) Memoryusage

The following table 5.1 shows time required for both algorithm.

Table I Comparison basis on time



IV. CONCLUSION AND FUTURESCOPE

A. *Conclusion*

Discovering association rules and frequent pattern mining have an ample space to propound deep and many efficient algorithms have been proposed up to now but applying those rules still found to be computationally expensive. An Apriori were suggested for frequent item set mining. Reverse Aprioriis partly refined in this study Instead of pairing all the frequent item sets at the end, which results in generating even higher number of generated candidate sets, it is viable to collect a heavy collection of frequent item set and then pruning it out thus giving lesser scans. And we have implement mapreduce to reduce search space and user constraint which effect minimum memory usage andtime.

B. *FutureScope*

Implementation of whole scenario with Hadoop framework which reduce time for execution and for frequent pattern use FP-growth algorithm which reduce database hit.

REFERENCES

[1]Carson Kai-Sang Leung, Christopher L. Carmichael "Efficient Miningof Frequent Patterns fromUncertainData" Seventh IEEE International Conference on Data Mining – Workshops DOI 10.1109/ICDMW.2007IEEE

[2] Carson Kai-Sang Leung,Richard Kyle MacKinnon,Fan Jiang "Reducing the Search Space for Big Data Mining for Interesting Patterns from Uncertain Data" 2014 IEEE International Congress on Big Data 978-1-4799-5057- 7/14

[3]C.K.-S.Leung&F.Jiang,"Frequentpatternmining fromtime-fadingstreamsofUncertaindata,"inDaWaK2011 (LNCS 6862), pp.252–264.

[4]C.K.-S.Leung&S.K.Tanbeer,"PUF-tree:Acompacttreestructureforfrequentpatternminingofuncertain data," in PAKDD 2013 (LNCS7818), pp.13–25.

[5]D.S. Rajput, R.S. Thakur, G.S. Thakur "Fuzzy Association Rule Mining based FrequentPatternExtractionfrom Uncertain Data"978-1-4673-4805-8/12 2012IEEE

[6]E. ¨O lmezo˘gullari& I. Ari, "Online association rule mining over fast data," in IEEE Big Data Congress2013, pp.110–117

[7]H. Yang & S. Fong, "Countering the concept-drift problem in big datausingiOVFDT," in IEEE BigData congress 13, pp.126-132.

[8] M.J.Zaki,"Parallelanddistributedassociationmining:asurvey,"IEEEConcurrency,7(4):14–25,Oct.–Dec. 1999.322.

[9]P. Agarwal, G.Shroff, & P. Malhotra, "Approximate incremental bigdataharmonization," in IEEE BigData Congress 2013, pp.118–125.

[10] S. Madden, "From databases to big data," IEEE Internet Computing, 16(3): 4–6, May–June2012.

[11] Yang & S. Fong, "Countering the

concept-drift problem in big data using iOVFDT," in IEEE Big Data Congress 2013, pp. 126–132.

[12] Mannila, H. Inductive databases and condensed representations for data mining. In International Logic Programming Symposium (1997), pp.21-30.

[13] Giannotti, F., and Manco, G. Querying Inductive Databases via Logic- Based User-De_ned Aggregates. In Procs. of the European Conference on Principles and Practices of Knowledge Discovery in Databases (September 1999), J. Rauch and J. Zitkov, Eds., no. 1704 in Lecture Notes on Arti_cial Intelligence, pp. 125{135.

[14] Giannotti, F., and Manco, G. Making Knowledge Extraction and Rea- soning Closer. In Procs. of the Fourth Paci_c-Asia Conference on Knowledge Discovery and Data Mining (April 2000), T. Terano, Ed., no. 1805 in Lecture Notes in Computer Science.