A hybrid algorithm combining weighted and hashT apriori algorithms in Map Reduce model using Eucalyptus cloud platform

[1]R. SUMITHRA, [2]SUJNI PAUL AND [3]D. PONMARY PUSHPA LATHA
[1]School of Computer Science, CMS College, Coimbatore, INDIA
[2]School of Information Technology, Al Dar University College, DUBAI.
[3]Department of Computer Applications, Karunya University, Coimbatore, INDIA

Abstract – Data mining performs a major role in data analysis which benefits enhancement of business. In modern era, distributed data mining becomes a major research area due to distributed computing. In distributed data mining data can be physically or virtually distributed that helps greatly in finding out interesting facts. Virtual distribution and distributed data mining gets its major impact on big data analysis with the help of cloud and grid computing. This research work performs data mining in a cloud environment using Eucalyptus platform with VMWare workstation and Hadoop platform. This work mines distributed data in a cloud environment using hybrid apriori association rule algorithm which combines the benefits of both hash-t and weighted apriori algorithms. Finally the work compares the performance with the existing implementation of weighted and hash-t algorithms.

Keywords - Weighted apriori; Association rule mining; Mapreduce; Hadoop; Euclyptus; Cloud; Data mining; HashT.

1 Introduction

This work starts with the analysis of various association mining algorithms especially apriori mining algorithms. And it finds out the fact that compared to traditional apriori, weighted and hash-t apriori perform well and explains its improvements with the help of literature study. It shows performance improves when implemented in a distributed environment. The main task of this work is to come up with a new modified algorithm in a Euclyptus environment which overcomes the drawbacks in normal weighted apriori and hash-t apriori algorithms. Results of the new implementation are being discussed. The paper concentrates on deriving a new apriori algorithm which performs well compared to others in cloud using eucalyptus nodes and with hadoop distributed file system and discusses the results obtained from new implementation in a cloud eucalyptus environment and how it overcomes the existing one.

2 Background study

Data mining is popular with various techniques such as classification, association, clustering, etc. Each technique is familiarized with numerous algorithms. To define the problem data should be distributed in a well planned way for distributed data mining. The interest of this research is moved to the direction of association rule algorithms which is suitable for market analysis and similar kinds of applications.

Apriori algorithm- Apriori is the best-known algorithm to mine association rules [1]. Breadth-first search is used to calculate the support of item sets and candidate generation function is used by exploiting the downward closure property of support. The distributed environment for the data mining process is discussed in paper [2]. Grid and Cloud are the latest technologies for the distributed nature. Cloud computing is comparable to grid computing, that is in grid computing unused processing cycles of all computers in a network are used to solve problems which are all impossible to be solved in any stand-alone machine [4].

Cloud computing has been evolved as the commercial process of Grid because the users can use the different types of services by paying [3] (Pay as you Demand and Use). Cloud computing enables the sharing of virtual resources in a secure and scalable manner. Hence cloud platform is well suited for data mining process as said in paper [6]. Hence a detailed study has been carried out in various algorithms in various cloud environments as said in various research papers [7].

Weighted apriori algorithm introduces weight which uses hubs and authorities as explained in

papers [10], [11] & [12]. The new research work proceeds with the weighted support which calculates weighted support based on the costs assigned to both items as well as transactions. The weighted algorithm finds out the rules that have a weighted support larger than a specified threshold [10]. The method also follows downward closure property. Hash-T apriori algorithm counts all the 1-itemsets for each transaction [5]. During the same pass all the possible 2 item sets in the transactions are hashed to a hash table. Hash table is used to reduce the candidate item sets. Using the support count calculated, the algorithm finds out the frequent item sets.

To implement the algorithms in a distributed environment map reduce model can be used [7]. The functions performed by Map reduce model are partitions the input, schedules the map reduce jobs among participating nodes, handles node failures and manages the required network communications. One implementation of Map Reduce is Apache Hadoop which gives an infrastructure for distributed computing and available as open source hosted by the Apache Software Foundation [8]. Hadoop is familiar for MapReduce and its Hadoop Distributed File System HDFS [4], Hadoop MapReduce is distributed data processing model and execution environment that runs on large clusters of commodity machines.

3 Previous work

In the background study conducted, it has been found that weighted and hash-t algorithms are performing well compared to other apriori algorithms, but so far they have been implemented in standalone machines. It has been found that it is better to have the same algorithms in distributed environment as discussed earlier.

So experiment has been started which is explained below. Hadoop is implemented with 4 nodes in VMWare virtual machine. Algorithms are coded in eclipse platform. Retail data set is retrieved from http://fimi.ua.ac.be/data/retail.dat. which is having 1,76,324 transactions and 16470 items. The execution is carried out for both the w-apriori and hash-t algorithms and comparison charts are given below. In weighted apriori, weight has been considered as a relevance of the item with other items while considering overall unique transactions. Minimum threshold is taken as .005. Hash node count is taken as 20.

The previous work has been discussed and results are compared in [14]. Execution time of both the algorithms gets improved because in normal apriori the process will happen in the single node. But in hadoop the process will happen parallel in the four nodes using map-reduce schema. Timing and Memory of weighted apriori is better than hash tree due to complexity. But the number of candidate set generation is less for hash tree as compared to weighted apriori. The hash tree has the complex execution complexity; therefore time and memory will be high. Though w-apriori performs well and gives more number of association rules, hash t gives better rules. Though hash-t gives better rules it occupies more memory and time.

4 Proposed work

Big data applied to datasets whose size is beyond the ability of commonly used software tools to capture, manage and process the data within a tolerable elapsed time.

Based on the previous work, it has been proposed to overcome the drawbacks of existing algorithms.

4.1. Weighted Apriori:

1.Itemset combination will be generated frequently. it will increase the candidate itemsets.
2.The weight computation for the each transaction will take more time to execute.
3. Less Accuracy
4. Not depent on data deviation

4.2. Hash Tree Apriori

1. High computational requirement
2. High memory utilization
3. Node processing requires high time to compute.

The benefits of both the algorithms could also not be bypassed, so that a new algorithm is invented with the positives of both weighted apriori and hash-t apriori algorithms.

To overcome the negatives and to combine the positive approaches of both the algorithms of previous work [14] a new tree-Based hybrid approach is being introduced in the proposed work.

4.3. Benefits of the hybrid approach

1. It will reduce the computation time as well as the accuracy of the frequent item prediction.
2. It will avoid the candidate set generation.
3. It will generate the tree structure and analyze their height, weight and reach ability for each node. [Reach ability refers to reach the end point of the transaction of each item in the tree]

4. The work definitely gives best outcome for the mining of frequent item sets with the state-of-art technologies like hadoop hdfs, vmware and eucalyptus platforms.
5. Such a combination of modified algorithm in hadoop VMWare environment doesn't exist earlier with respect to the literature survey.

Before discussing the new algorithm few points are to be explained. Frequent Item mining is the process of predicting the frequent patterns in the dataset. It will be applicable to real-time data such as Gene/Protein Sequence, Public usage, etc. Here prediction of combination and relationship between items is the most important to predict the association between the items. In the proposed system the selected datasets (Mushroom & Retail) contain numerous numbers of transactions. From these transactions first it is needed to predict the items in the transactional database. For predicting the frequent items and association rules in the transactional database it generally consumes lot of memory storage and time. To overcome the problem the new proposal computes the frequent pattern without the generation of candidate item sets. Tree based mining approach is used to reduce the consumption of the memory usage in the association rule prediction during reducing phase. It uses correlation based approaches to optimize the process.

4.1. Hybrid Apriori

The hybrid algorithm mines the transaction database which is considered to be the data for distribution. It finds the frequent item sets in the first iteration and also finds the relationship matrix. For each transaction it scans the database for counts and obtains the subset of transaction that is the candidate. For each candidate, count is increased. Then using the relationship matrix frequent tree is constructed. If the tree is having a single transaction path then for each combination (which is there in combination tree) of the nodes in the transaction path it generates pattern and that pattern with support count more than the minimum support count of nodes already in the combination tree, will be stored as a result tree. The same manner is followed throughout the construction of combination tree and result tree and the process is iterated.

The flow diagram of the proposed work is given in Fig.1.and Fig.2.

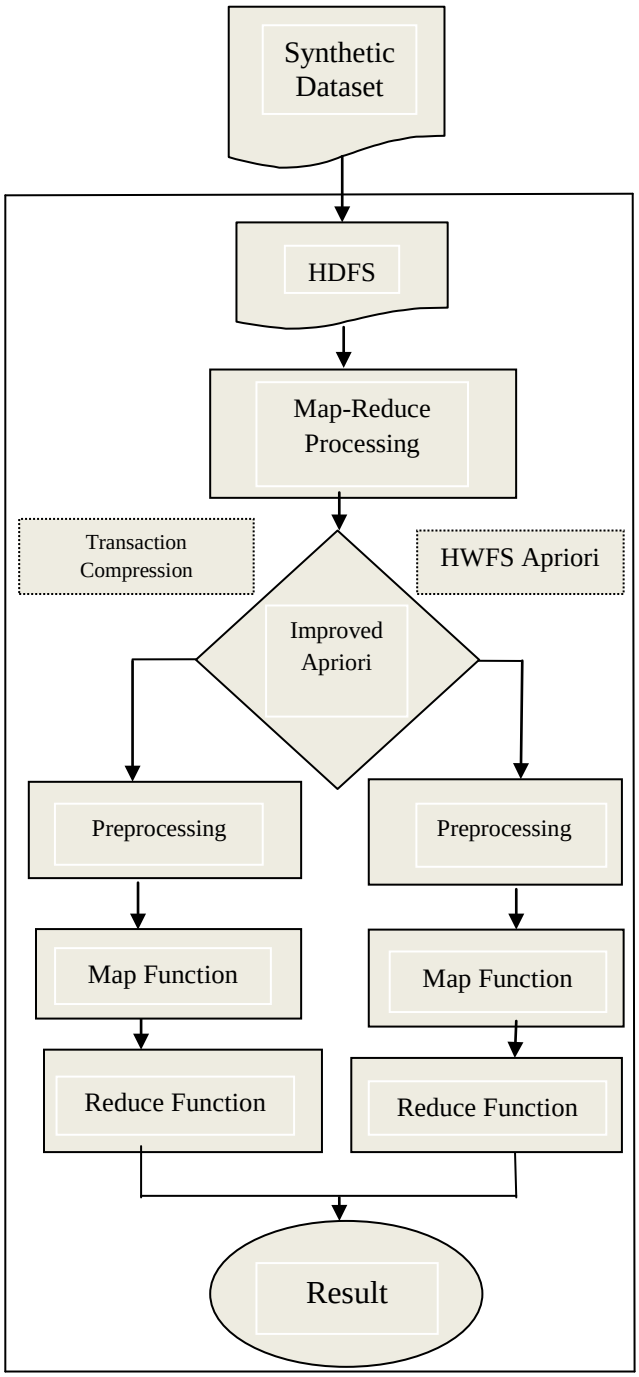

Fig.1. Flow diagram1.

The HWFS (H stands for Height, W stands for Weight, F stands for Frequency and S stands for support) Apriori in the figure.1. refers to the concept of finding Height, Weight, Frequency & Support in same apriori tree. The flow diagram given above clearly depicts the map and reduce functions are working in a distributed environment with tree based mining approach to reduce the consumption of the memory usage in the association rule prediction.

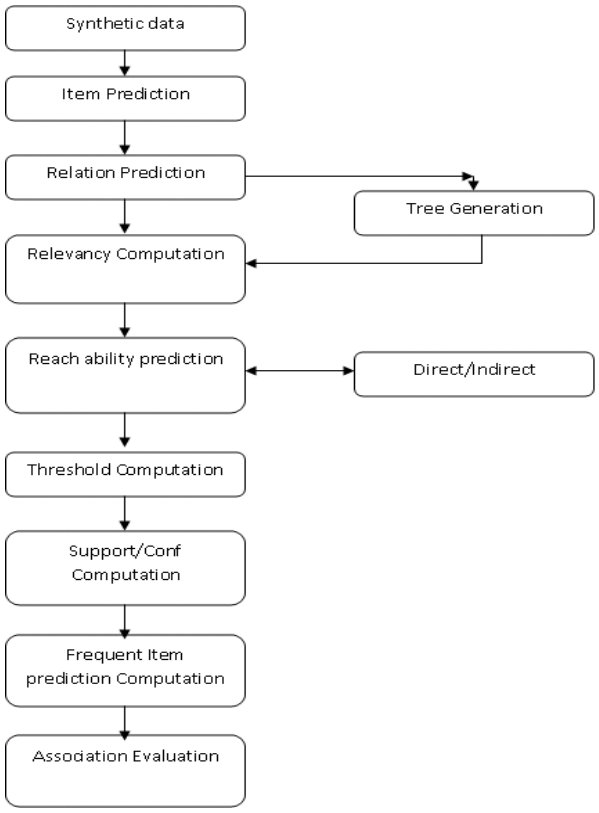

Fig.2. Flow diagram2

The diagram in fig.2. depicts the way of representing the logic of the tree formation. First the process starts with item prediction which finds the frequency of items, secondly relation prediction is done and a relationship matrix is formed. Then reachability prediction is being done to find the direct and indirect relationship using the matrix. Support/confidence are computed which is the weighted support and confidence and can be used in association evaluation to find the association rules.

4.2. Taxonomy Of Hybrid Apriori Algorithm

The explanations for the steps given in flowdiagram2 are depicted as follows.

- **Item Prediction** is the process of predicting the distinct items in the synthetic dataset. And also to find the items frequency of the given dataset.
- **Relation Prediction** process finds the association between the items in the frequent mining. This will help to formulate the tree based frequent mining approach. This tree will contain the relationship with direct and indirect connections.
- **Relevancy Computation** is the process of predicting the relationship between the nodes in the tree which means items in the frequent item tree. This will be formulated by the matrix computation.
- **Reach ability prediction** is identified in the tree of frequent items between the nodes. This shows both direct and indirect relationship between the nodes in the tree.
- **Support/Conf Computation,** with Support and confidence of the tree items in the frequent mining the tree height and weight based support and confident are calculated. This approach will reduce the iterations in the frequent item prediction.
- **Association Evaluation** is the process of predicting the association rules in the frequent items synthetic dataset.

4.3. Pseudo code

The pseudo code of the new algorithm is depicted below.

Input: D = Synthetic Frequent Item Dataset
Output: Frequent Item and Associations
Procedure
$\boldsymbol{T_L}$ **= Read_TransactionalDataBase(D);**
$\boldsymbol{F_i = \emptyset}$ **;**
$\boldsymbol{FI = \emptyset}$ **;**
//Compute Frequency
For t = 1 … n **Then** // n = Number of Transactions
$I_L = t.\boldsymbol{getItems}()$;
$\boldsymbol{For}\ i = 1 \ldots m\ \boldsymbol{then}$
If $(FI.\boldsymbol{Contains}(I_i))$ **then**
$F_i = 1$;
$\boldsymbol{Else}$
index = $FI.\boldsymbol{Indexof}(I_i)$;
$F_{I_i} = F_{I_i} + 1$;
$\boldsymbol{End\ If}$
End For i
End For t
Tree $\boldsymbol{T_r}$ = **generateTree**(F_i);
For t = 1 … n **Then** // n = Number of Transactions
if $\boldsymbol{T_r}$.contains($\boldsymbol{T_t}$) then
For each combination (denoted as CT) of $\boldsymbol{T_t}$
Generate pattern CT U RT with $\boldsymbol{SC_t}$ =$\boldsymbol{min}(\boldsymbol{SC_t})$;
Else for each $\boldsymbol{N_i} \in \boldsymbol{T_t}$
header of Tree{
generate pattern CT=$\boldsymbol{N_i}$ U RT with $\boldsymbol{SC_i}$ = $\boldsymbol{N_i.SC_i}$
construct CT's conditional pattern base and then CT's conditional Frequent TreeCT;

```
if TreeCT != 0 then

RT = CT;
Repeat;
}
End IF
End For t
```

5 Optimization

The ways for improving the algorithm is thought over. Any algorithm will give better performance if mathematical model is used in a suitable way. In this proposed algorithm it is found out that

performance can be improved by using a correlation model in the tree construction phase. The formula is given below. Correlation between Trees is

$$\boldsymbol{Cr}(\mathrm{CT},\mathrm{RT}) = \frac{N\sum_{k=0}^{n} x^{CT} y^{CT} - \sum_{k=0}^{n} x^{CT} \sum_{k=0}^{n} y^{RT}}{\sqrt{N\sum_{k=0}^{n} x^{CT\,2} - \left(\sum_{k=0}^{n} x^{CT}\right)^2 * N\sum_{k=0}^{n} y^{RT\,2} - \left(\sum_{k=0}^{n} y^{RT}\right)^2}} \quad (1)$$

x^{CT} - Candidate Conditional tree element values, y^{RT} – Result tree element Values, N - Number of elements in the tree

Data mining in a distributed environment surely has to prove the data integrity before and after the distribution, whether any integrity and security violations have been done to the data. In this algorithm this process has been done using Hash Message Authentication Code[9] and is proved to be correct.

6 Computing environment

The new algorithm is implemented in java using eclipse platform. Eucalyptus is being used to create the cloud platform and eucalyptus nodes have been created in VMWare machines. It will help to find the results of the frequent item mining in the real cloud environment. This improves the performance of the new implementation. Another VMWare machine has been set up for hadoop distributed file system (HDFS). Here hadoop will be used as the executing node in the eucalyptus environment to handle the mining process. Two datasets Retailitem and Mushroom have been taken from UCI data repository. The hadoop environment here is having 4 nodes which are handling the datasets using HDFS. Coding has been executed and result is obtained.

7 Results

The results are discussed in terms of frequent item generation, memory utilization, time consumption. The new hybrid apriori algorithm outperforms weighted apriori and hash-t apriori algorithms in terms of time, memory, etc. The comparison is carried out with the implementation of weighted and hash-t aprioi algorithms discussed in the previous work [11] and with the new modified apriori algorithm. Weighted apriori stops in 6 iterations, Hash-t apriori stops in 5 iterations whereas Hybrid apriori stops in just 4 iterations and gives better frequent item set.

The graphs in figure [3] show the memory usage of three algorithms (Weighted, HashT & Hybrid Apriroi).

The new algorithm uses only around 50Mb because of tree construction phase instead of candidate generation at each stage.

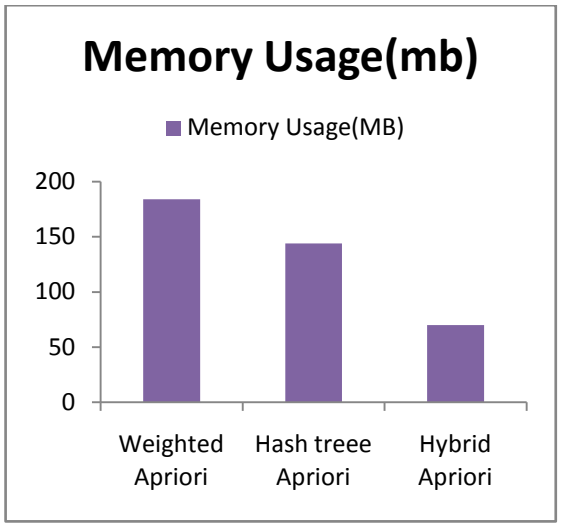

Fig. 3. Memory Usage comparison

In the same way figure [4] shows the time consumed by the three algorithms and shows that the hybrid algorithm consumes less time compared to others.

The graph in fig [5] shows the consolidated performance of all the discussed algorithms in this work in terms of item size, memory and time. Obviously Hybrid algorithm works well. Here in this graph PApriori refers to hybrid new apriori algorithm of the research work.

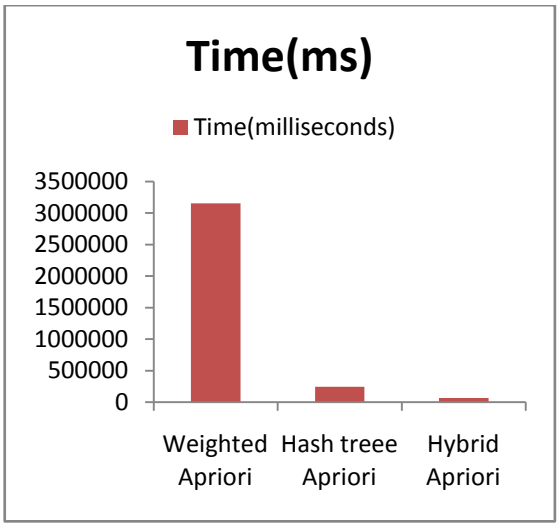

Fig.4. Time comparison

To find out the performance of the algorithm a detailed study has been done without hadoop and with hadoop environment.

Fig.5. Execution Evaluation

The Table.1. shows the difference in terms of time (microseconds). It clearly depicts how the new hybrid algorithm works without and with hadoop and eucalyptus platforms. The readings shown in last column of the table shows that when the algorithm is implemented in hadoop and eucalyptus platform time is considerably reduced.

The result achieved here gives a breakthrough in the research work of distributed data mining, as the big data of today business world cannot be kept in a single place and mined. Surely data have to be distributed at various nodes and mined. The new algorithm discussed in this research work gives a better direction in the field of cloud data mining.

Table 1 Comparison Table

Dataset Size (MB)	Without Hadoop (ms)	With Hadoop (ms)	Without Eucalyptus (ms)	With Eucalyptus (ms)	Hadoop+ Eucalyptus (ms)
100	1500	1200	1500	1300	1000
200	2000	1700	2000	1600	1100
300	2412	1965	2412	1756	1200
400	2895	2345	2895	2156	1800
500	3342	2812	3342	2689	1956
600	3665	3014	3665	2895	2189

8 Conclusion

This work concludes on implementing a new improved weighted hashT apriori algorithm, which is being termed as a hybrid algorithm, in a hadoop - map reduce environment and obtains results using a retail data set and mushroom dataset. Various comparison charts are given to analyse the performance and it is made sure that the new one works well in terms of time and memory and gives importance to weighted support and tree based approach to reduce the candidate set generation. Optimization and data integrity are also taken care which are all important in the success of any algorithm especially distributed cloud data mining. The best algorithms in association rule mining has been chosen which gives better insight in the work but in future the work can be enhanced by having different sort of algorithm, with optimum tree pruning methods which will eliminate the not frequently used items in a better way. The research work may further proceed to highlight the importance for still better tree construction. a research work could not be confined to fixed number of distributed nodes. So the same work can be tested with 8 and 16 number of nodes. Obviously the coding will work and give better results. Next phase of modification in the algorithm can be done in the tree construction step to give a more optimized tree.

References

[1]Agrawal, R. & Srikant. R. Fast algorithms for mining association rules. *Proceedings of the 20th VLDB Conference Santiago, Chile (VLDB '94)*, 1994,pp: 487-499.

[2]Aouad, L.M., et al., . Distributed frequent itemsets mining in heterogeneous platforms. *J. Eng. Comput. Archtecture*: 1(2), 2007.

[3] Bhagyashree Ambulkar. & Vaishali Borkar.,. 'Data Mining in Cloud Computing', *IJCA*, 2012

[4] Domenico Talia., 'Grid-based Distributed Data Mining Systems, Algorithms and Services', *SIAM Data Mining Conference*, 2006.

[5] Grudzinski, P. & Wojciechowski. M. , Integration of candidate hash trees in concurrent processing of frequent itemset queries using apriori. *Control Cybernetics*.:38(1), 2009.

[6] Kambatla, K., et al., . Towards optimizing hadoop provisioning in the cloud. Proceedings of the 2009 Conference on Hot Topics in Cloud Computing *(HotCloud'09)*: 2009,Article No. 22.

[7] Juan Li , Pallavi Roy , Samee U.et.al.. Data mining using clouds: An experimental implementation of apriori over mapreduce. *Proceeding of the 12th IEEE International Conference on Salable Computing and Communication (ScalCom 2102)*, 2012,Changzhou, China.

[8] Robert Grossman. & Yunhong Gu.,. 'Data Mining Using High Performance Data Clouds: Experimental Studies Using Sector and Sphere', *NSF*, 2004.

[9] Federal Information Processing Standards Publication, The Keyed-Hash Message Authentication Code (HMAC), *Information Technology Laboratory, National Institute of Standards and Technology, Gaithersburg,* MD, 2001.

[10] Sun, K. & Bai.F. . Mining weighted association rules without preassigned weights. *IEEE T. Knowl.Data En., 20(4):* 2008, 489-495.

[11] Wang, K. & Thomas Su.M.Y.. Item selection by hub-authority profit ranking. *In SIGKDD, 2002*, pp: 652-657.

[12] Wang, W., Jeong yang & Philip.S., Efficient mining of weighted association rules. *KDD 2000, Boston, MA USA*, 2000, pp: 270-274.

[13] Zhou zhao, Da Yan & Ng, W., Mining probabilistically frequent sequential patterns in large uncertain databases., *IEEE transactions on knowledge and data engineering,* 2014, vol. 26, no. 5.

[14] R.Sumithra, Sujni Paul and D. Ponmary Pushpa Latha, Apriori Association Rule Algorithms using VMware Environment. *Research Journal of Applied Sciences, Engineering and Technology*, 8(2): 2014, 160-166. .