Volume 5, Issue 5, May 2015 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

An Approach to Quantative Analysis of Apriori and Eclat Algorithms using R-Programming

Tanu Jain*
Amity University
Jaipur, India

Dr. A. K Dua
HOD (CSE), Amity University
Jaipur, India

Varun Sharma
Amity University
Jaipur, India

***Abstract*— Apriori and Eclat algorithms are the most widely used algorithms in the area of association rules mining used for mining of frequent item sets and to discover associations between them. R is a domain specific language for data analysis and analytics. It is already being used across different disciplines from Computer Science to Social Sciences.**

In this research a qualitative and quantitative analysis of Apriori and Eclat algorithms is done using R Environment. Different R-Packages and libraries are used for the access of different datasets and their connectivity with R. Both algorithms are performed using different data sets and are further analyzed on the basis of their performance. The performance analysis is based on different performance matrices including support, lift speedup etc. in order to identify their quantitative performance with different volume of datasets.

***Keywords*— Apriori, Eclat, Association Rule Mining, R language, R Environment**

I. INTRODUCTION

Association rule learning is one of the most popular and widely used process in order to find the interesting relations among variables within large databases and datasets. We always need to find strong rules to be discovered in large databases using different measures of performance matrices. Quantitative Analysis simply is the survey of a event, mostly a financial market, by means of complex mathematical and statistical modelling as per the standard definition.

A. Apriori Algorithm

This algorithm has been very frequently used for mining of frequent item sets and to discover associations. The major difference in Apriori is the less candidate itemsets it generates for testing in every database pass. The search for association rules is guided by two parameters: support and confidence.[1]

It is a breadth-first search algorithm. Apriori returns an association rule if its support and confidence values are above user defined threshold values. The output is ordered by confidence. If several rules have the same confidence then they are ordered by support. Thus Apriori favors more confident rules and characterizes these rules as more interesting.

B. Eclat Algorithm

Eclat generates less number of sequence tables which takes less time for generation of frequent accessed patterns as compared to Apriori. In apriori if massive data is their then it takes huge time to generate the frequent accessed patterns. Eclat implementation represents the set of transactions as a bit matrix and intersects rows give the support of item sets. It follows a depth first traversal of a prefix tree.

C. R-Programming

R is a domain specific language for data analysis, which is, according to some measures, the most popular such platform. The language syntax has roots in the 1970's at Bell Labs, and was developed specifically as a language for statistical data analysis.

R is a free, open-source language and importantly, is already used widely across disciplines (from Computer Science to Social Sciences - such as Political Science). As such, providing freely accessible software to employ classical algorithm systems as part of an R package will allow practitioners and researchers from a wide range of areas to make use of the research developed by the apriori and éclat research community who in turn can incorporate the potentially vast amount of feedback and real-world deployment information into their research.

II. LITERATURE SURVEY

The process of discovering the frequent item sets within a set of transactions is a well-known method for a problem simply known as market basket analysis, this is used in order to find regularities or frequent patterns from the shopping habits of consumers of shopping portals, supermarkets, companies that supports mail-order, on-line shopping sites etc. In particular, it is intended to identify the sets of products that are bought together on a regular basis.

The major problem of identifying the frequent item sets, or the item sets that exists in a user-specified transactions, is that there are various possible sets, which renders naıve approaches ineffective because of their expensive execution

time.[1] Among all these most popular and sophisticated methods two algorithms known under the names of Apriori and Eclat are widely used. Both algorithms functions on a top down searching approach in the subset lattice of items.
A simple example of a subset lattice for five different items is given in figure 1 (empty set are not considered).

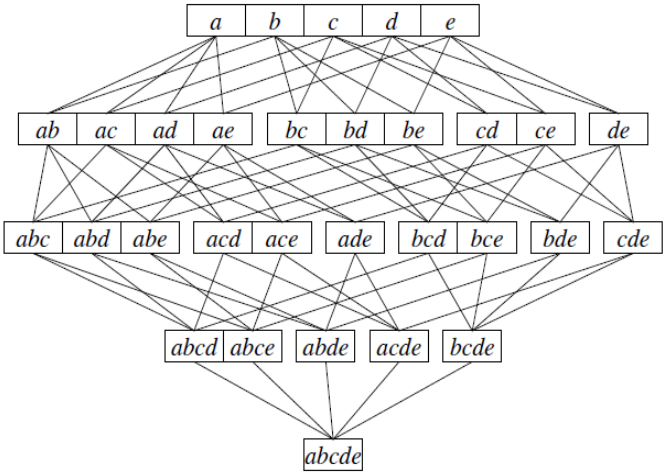

Fig. 1: A subset lattice for five items

The edges in above figure indicate subset relations among various item sets. For the formation of the search, these both algorithms organizes the subset lattice in the form of a tree which is prefix in nature, that is displayed in Figure 2 for five items. In this tree those item sets are added in a single node that is having the same prefix with respect to some arbitrary, but fixed numbers of order of the items (within the five items example, the order is a, b, c, d, e simply).[6]

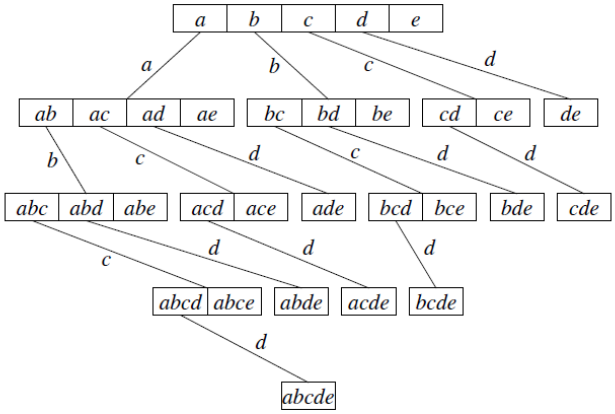

Fig. 2: A prefix tree for five items

Along with the same structure, the item sets are consists within a single node of the tree that can be constructed in the given manner easily:

- In first step take all the items in which the edges are leading to the nodes that are labeled and add the item that succeeds, in case of fixed orders of the items, the last edge is labeled on the path.
- It should be noted that in the similar way we require only a single item to differentiate between the items represented in a single node, this is much relevant for the implementation of both these algorithms.[6]

The major differences between Apriori and Eclat is the approach that how they traverse the prefix tree and how they calculate the *support* for an item set, that is the total number of transactions the item set contained within.
Apriori basically traverses the displayed tree in BFS(breadth first order), it means it first checks for the itemset of size 1 and after that further for the item set of size 2 etc. Apriori assesses the support of itemsets may be by checking each of item set which the transactions contains within, or by traversing for a transaction every subset of the most recently processed size and increasing the related item set counters.[6] This approach is normally preferable.

Eclat, navigates the prefix tree top to bottom first request, being inverse to apriori. It simply means that, it extends an item set prefix until and unless it reaches to the boundary in between the infrequent and frequent item sets and then further backtracks to process the coming prefix. Eclat calculates the support of every single item set by creating the list of all the identifiers of transactions that contain the item sets. It uses the approach of intersecting two lists of transaction identifiers for two different item sets only by a single item or together form the item set recently processed.

A. The R Programming

R is popular among organizations as a statistical and programming language. It is a software environment used in graphics and data analysis. The R language is mostly used among the statisticians and data scientist for the development

of statistical applications and analysis of data.[7] R is a free and open source software environment used for statistical computing. It compiles and executes on a wide range of UNIX based operating systems, Windows based and Mac based operating systems.[8]

In the past decade, the momentum coming from academia as well as from industry has settled the R programming language as one of the most important tool for data analysis, computational statistics, data visualization and for data science.

From all over the world, millions of data scientists and statisticians uses R to solve their most challenging problems related to data in fields ranging from computational biology, data science to quantitative marketing analysis.

R is one of the most popular language for data science as well as an essential tool for Finance analysis and analytics-driven companies such as Google, Facebook, and LinkedIn.

B. Why R?

1. R is free software. R is an official GNU project and distributed under the Free Software Foundation General Public License (GPL).
2. R is a powerful data-analysis package with many standard and cutting-edge statistical functions. See the Comprehensive R Archive Network (CRAN)'s Task Views to get an idea of what you can do with R.[8]
3. R is a programming language, so its abilities can easily be extended through the use of user-defined functions. A large collection of user-contributed functions and packages can be found in CRAN's Contributed Packages.
4. R is widely used in political science, statistics, econometrics, actuarial sciences, sociology, finance, etc.
5. R is available for all major operating systems (Windows, Mac OS, GNU-Linux).
6. R is object-oriented. Virtually anything (e.g., complex data structures) can be stored as an R object.
7. R is a matrix language.
8. R syntax is much more systematic than Stata or SAS syntax.
9. R can be installed on your USB stick.

III. PROBLEM STATEMENT

Apriori and Eclat both are the well-known and highly used algorithms. We need to find that which one better in which scenario? When to apply Apriori algorithm and when to apply Eclat algorithm while keeping their performance at their best. Which algorithm should be apply to achieve the various requirements of association rules mining that includes confidence, support, lift and other performance matrices.

IV. EXPERIMENTAL SETUP & APPROACH

For the implementation of Eclat & Apriori algorithms we've used the R Environment with various user defined and system defined libraries.

A. Setting up R-Environment:

We need to install required packages in R environment:

User Libraries:

- arules
- arulesViz

System Libraries:

- matrix
- grid

Data Sets:

- We'll use Data sets consisting of thousands transactions and different items with RDATA extension. RDATA extension supports R-Environment.

Finding Association Rules:

1. We'll use Apriori() and Eclat() functions from arules package to mine the association rules from the datasets, While setting the Parameter specification and algorithmic control.
2. We'll inspect the rule and select from them as per the required parameter specifications we want.

Removing Redundancy:

1. Using the matrix package we'll first find those rules which are the subset of another rules.
2. Remove all redundant rules.

Interpreting Unique Rules:

1. We'll inspect all the unique rules and select from them as per the required parameter specifications.

Results and Visualization:

1. We'll use the different plotting method of arulesViz library to visualize the rules.
2. Plot() method will visualize all the rules.
3. Grouped() method give the grouped matrix for all the association rules.
4. Graph() method creates the different kinds of graphs for the association rules as per the different control parameters.
5. Paracoords() methods gives the different parallel coordinated for the association rules.

B. Performance Matrices

To choose fascinating standards from the arrangement of every conceivable rule, requirements on different measures of interests and significance can be utilized. The best-known limitations are least edges on support and confidence.[5]

- The *support* $\operatorname{supp}(X)$ of an itemset X is characterized as the extent of exchanges in the database which contain the itemset.
- The *confidence* of a rule is defined as $\operatorname{conf}(X \Rightarrow Y) = \operatorname{supp}(X \cup Y) / \operatorname{supp}(X)$.
- The *lift* of a rule is defined as $\operatorname{lift}(X \Rightarrow Y) = \frac{\operatorname{supp}(X \cup Y)}{\operatorname{supp}(X) \times \operatorname{supp}(Y)}$ or the proportion of the support to that normal if X and Y were autonomous.
- The *conviction* of a rule is defined as $\operatorname{conv}(X \Rightarrow Y) = \frac{1 - \operatorname{supp}(Y)}{1 - \operatorname{conf}(X \Rightarrow Y)}$.

V. CONCLUSION

In this paper, study includes depth analysis of algorithms and discusses some problems of generating frequent itemsets from the algorithm.

This paper will adopt efficient sequential pattern mining techniques using the Apriori and Eclat algorithm for the filtered data set. Both the algorithms helps to find out the navigation behavior of the user based on the previous visits and also shows the comparison of the two techniques adopted for predicting user access behavior using R-programming. Discovering the frequent itemsets from the two algorithms a also shows that Eclat algorithm serves better for the large databases despite Apriori as it generates less tables and therefore less time it takes to perform the analysis.

REFERENCES

[1] Rahul Mishra et. al. "Comparative Analysis of Apriori Algorithm and Frequent Pattern Algorithm for Frequent Pattern Mining in Web Log Data." (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 3 (4), 2012, Pp. 4662 – 4665.

[2] O. R. Zaiane, J. Han, and H. Zhu. Mining Recurrent Items in Multimedia with Progressive Resolution Refinement. ICDE'00, 461-470, San Diego, CA, Feb. 2000

[3] Sathish Kumar et al. "Efficient Tree Based Distributed Data Mining Algorithms for mining Frequent Patterns" International Journal of Computer Applications (0975 – 8887) Volume 10– No.1, November 2010.

[4] Haoyuan Li,Yi Wang,Dong Zhang, Ming Zhang,Edward Chang 2008."Pfp: parallel fp-growth for query recommendation Proceedings of the 2008 ACM conference on Recommender systems Pp. 107-114.

[5] Sachin Sharma, Vidushi Singhal and Seema Sharma, "A SYSTEMATIC APPROACH AND ALGORITHM FOR FREQUENT DATA ITEMSETS", Journal of Global research in computer science, Volume 3, No. 11, November 2012.

[6] Christian Borgelt,"Efficient Implementations of Apriori and Eclat", Department of Knowledge Processing and Language Engineering School of Computer Science, Otto-von-Guericke-University of Magdeburg Universitatsplatz 2, 39106 Magdeburg, Germany

[7] Fox, John and Andersen, Robert (January 2005). "Using the R Statistical Computing Environment to Teach Social Statistics Courses", Department of Sociology, McMaster University. Retrieved 2006-08-03.,http://en.wikipedia.org/wiki/R_%28programming_language%29

[8] The R Project for Statistical Computing**,** http://www.r-project.org/

