KIET International Journal of Intelligent Computing and Informatics,
Vol. 1, Issue 1, January 2014

Implementation of Apriori Algorithm using WEKA

Ajay Kumar Shrivastava
Associate Professor
KIET Group of Institutions, Ghaziabad
ajay@kiet.edu

R. N. Panda
Associate Professor
KIET Group of Institutions, Ghaziabad
rabi.panda@kiet.edu

Abstract—In this current fast moving world, information is the most common feature in every aspect of the life. It can be used to perform analysis and it helps in decision making. But due to huge collection of datasets the analysis and extraction of useful information from the database, creates a problem. Association rules have been used to extract the useful information from the large databases. Apriori algorithm is one of the most useful algorithm for the association rule mining. In this study the implementation of the Apriori algorithm using WEKA has been explained. A new dataset for this study has been created and tested using the ARFF files.

Keywords- Apriori algorithm; association rules; data mining; Weka.

I. INTRODUCTION

In this current world, globalization is the main feature of any environment. Everyone has to be update, fast and forward and information is the main element for it. For survival in this world it's the basic need to use and to store the information means to prepare a proper database or dataset to analyze.

Using and storing the database is not an issue, but finding the relevant dataset or to analyze the meaningful dataset for a particular aspect, from the junkyard of the database is very big problem in analysis of a specific part of the database.

To solve this problem the concept of data mining is used to abstracts the desirable information. Useful information from the large databases has been extracted in the form of the association rules. There are many algorithms have been developed to extract the association rules from the large databases. Apriori algorithm is the most popular algorithm to extract the association rules from the databases [1].

To implement the Apriori algorithm, there are many tools available in the market. WEKA is a open source software tool for implementing machine learning algorithms [2]. In this study, the dataset has been created in the form of ARFF files and tested using the Apriori algorithm in WEKA software tool.

II. WEKA [WAIKATO ENVIRONMENT FOR KNOWLEDGE ANALYSIS]

WEKA is the collection or a suite of the tools for performing data mining with the implementation of the 'association rules' in it. Basically it is a collection of machine learning algorithm for the task of data mining, which is able to be applied directly to dataset or can call from your own java code.

It is collection or suite of tools for performing the - data preprocessing, classification, regression, clustering, association rules and visualization type operations and it also can be enhance any new machine learning scheme. In this study the WEKA 3.6.5 has been used. There are following tools available in Weka.

Explorer is used for exploring and extracting the dataset on which the operations has to be performed. Experimenter is used to perform experiments or statistical tests on the dataset. Knowledge Flow provides same functionalities as provided by Explorer but with a drag-and-drop interface. It helps in incremental learning. Simple CLI provides simple Command Line Interface that allows direct execution of Weka commands for Operating System that do not provide their own command line interface [3].

WEKA is performing its operations on the concepts of the association rules of data mining.

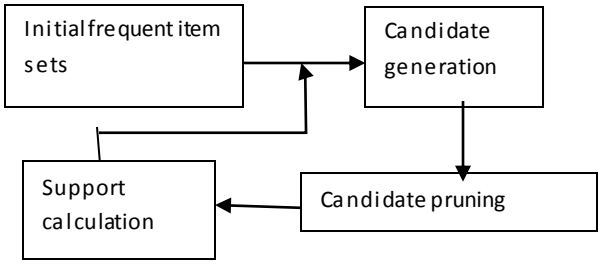
III. ASSOCIATION RULES

Association rules are if/then statements that help uncover relationships between seemingly unrelated data in a relational database or other information repository. An example of an association rule is "If a customer buys a dozen eggs, he is 80% likely to also purchase milk". An association rule has two parts, an antecedent (if) and a consequent (then). An antecedent is an item found in the data. A consequent is an item that is found in combination with the antecedent.

Association rules are created by analyzing data for frequent if/then patterns and using the criteria support and confidence to identify the most important relationships .They are divided into separate categories in the data mining and used in the Weka to perform the operations.

IV. APRIORI ALGORITHM

The apriori algorithm is a popular and foundational member of the correlation based 'Data Mining kernels' used today. It is used to process the data into more useful forms, in particular, connections between set of items.
The apriori algorithm is divided into 3 sections as-



Initial frequent item sets are fed into the system, and candidate generation, candidate pruning, and candidate support is executed in turn. The support information is fed back into the candidate generator and the cycle continues until the final candidate set is determined. A frequent item set is a set of one or more items that often occur in the database one item, and often occurs together in the same basket within the database if it consists of more than one item. The cutoff of how often a set must occur before it is included in the candidate set is the support.

The general approach is to implement the Apriori algorithm in the most efficient manner possible, utilizing a minimum of hardware and a minimum of time, as well as insuring that utilization of the hardware comparators in near 100% [4]. The algorithm is as follows

- Join Step: C_k is generated by joining L_{k-1} with itself.
- Prune Step: Any (k-1) item set that is not frequent can't be a subset of a frequent k-item set.
- Pseudo-Code:

C_k : Candidate item set of size k

L_k : Frequent itemset of size k

L_1 = {frequent items}

For(k=1;L_k!= ϕ;k++) do begin

C_{k+1}=candidates generated from L_k;

For each transaction t in database do

Increment the count of all candidates in C_{k+1} that are

Contained in t

L_{k+1}=candidates in $C_{k+1}$with min_support

End

Return $\cup_k L_k$;

After applying the Apriori algorithm on the dataset given in table 4.1 the three items are associated with each other having support value of 2.

Table 4.1: Dataset

T_ID	Items
100	1 3 4
200	2 3 5
300	1 2 3 5
400	2 5

V. IMPLEMENTATION

In Weka, basic implementation had done on the ARFF (Attribute-Relation File Format) files. An ARFF file is an ASCII text file that describes a list of instances sharing a set of attributes. ARFF files were developed by the Machine Learning Project at the Department of Computer Science of The University of Waikato for use with the Weka machine learning software.

An ARFF file is an ASCII text file that describes a list of instances sharing a set of attributes. ARFF files have two distinct sections. The first section is the Header information, which is followed the Data information [2].

Lines that begin with a % are comments. The @RELATION, @ATTRIBUTE and @DATA declarations are case insensitive.

i. Header part: The Header of the ARFF file contains the name of the relation, a list of the attributes (the columns in the data), and their types.

The relation name is defined as the first line in the ARFF file. The format is:

@relation <relation-name>
where <relation-name> is a string. The string must be quoted if the name includes spaces.

Attribute declarations take the form of an orderd sequence of @attribute statements.

@attribute <attribute-name> <datatype>

Example-

@relation Computer

@attribute T_id {100,200,300,400}
@attribute Num1{0,1}
@attribute Num2{0,1}
@attribute Num3{0,1}
@attribute Num4{0,1}
@attribute Num5{0,1}

ii. Data part:The @data declaration is a single line denoting the start of the data segment in the file. The format is:

@data

Example-
@data
100,1,1,1,0,0
200,1,1,0,1,1
300,1,0,1,1,0
400,1,0,1,0,0

iii. Creation of an ARFF file: Define and create the ARFF header part and the data part of the file along with define its attributes and relations in any notepad program and simply save it into .arff file format.

To view that ARFF file-
a) Open Weka and select its tools option from menu.
b) Select the option arff viewer from pop-up menu.
c) Choose the file option from the arff viewer window.
d) Open the created arff file from the location.

After opening the arff file from arff viewer it is shown in fig.5.1.

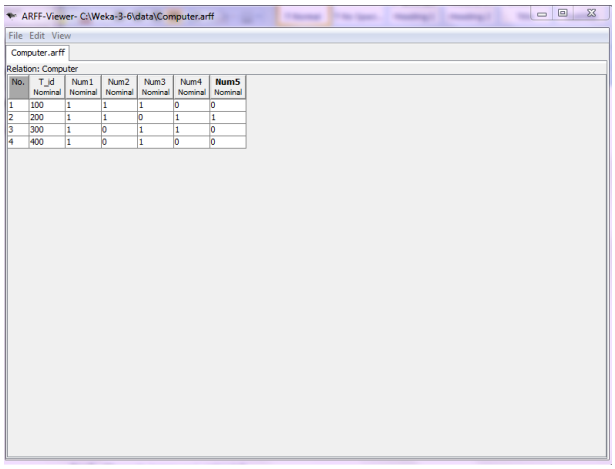


Figure 5.1: Scrren shot of ARFF viewer.

iv. Experimental steps to perform the test on the problem:

Step1:- Open the Explorer application of the WEKA tool.

Step2:- Choose its 'preprocess' option/tab.

Step3:- now click on 'open file' button to choose any 'arff' file which have to be implemented. And set the path for it to fetch that file.

Step4:- After include/fetch the arff file open the 'Associate' tab of the Explorer window.

Step5:- click on the choose button to select the algorithm to implement (as we take here the Apriori Algorithm).

Step6:- click on start button to start the working the operation of the algorithm on the selected arff file.

Step7:- the result of this operation will be display in the 'Associator output' box in the explorer window.

VI. OBSERVATIONS AND RESULTS

When the apriori algorithm was implemented on the Computer.arff file, it produced best association rules for that particular computer.arff file or dataset. Now the observation took place, to observe the result of the same operation, on the same dataset by the use of weka tool and made the comparison between both cases.

i. Open the file in preprocess segment: When the chosen ARFF file is open in the Preprocess section, there were two boxes appeared as in fig.6.1.

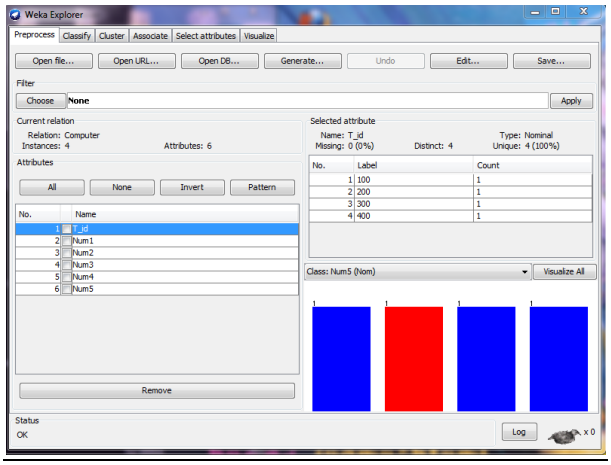


Figure 6.1: Screen shot of Weka Explorer (Preprocess).

left most box was the 'Attributes'- which showed the attributes name of that particular arff file which was currently opened and provide the checkboxes to select the attributes and there was a button to remove the checks.

Right side box was 'selected attributes'- which upper portion showed the value items of the selected attributes with the description of their labels and their counts and other information. And the lower portion showed the graphical representation of the attributes with itself.

ii. Open the file in Associate segment: When the chosen ARFF file is open in the Associate section, there were an output box appeared as in fig.6.2.

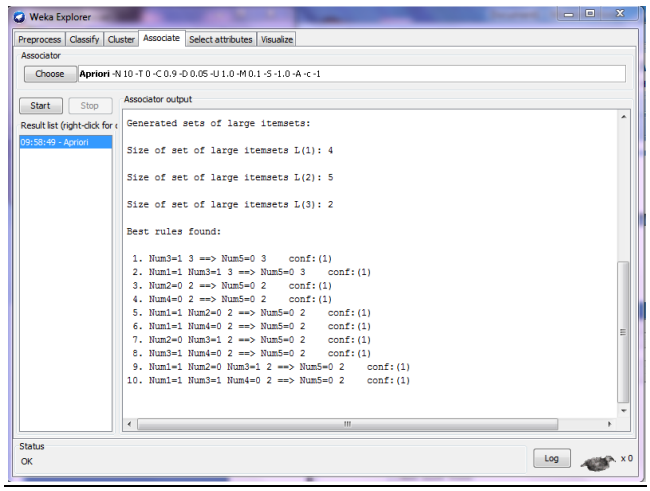


Figure 6.1: Screen shot of Weka Explorer (Associate)

In this segment there were option to chose the Algorithm to be implemented on the selected arff file, and then had to press the start button to find the associator output. In that segment the chosen algorithm was Apriori algorithm. Click on Apriori textbox and a box will open .Then turn car option to true from 'false' it was done because If car option was enabled, class association rules are mined instead of (general) association rules.. And after pressing the start button the associator output Box showed the result of it which gave the some information in the form of 10 best association rules as in fig.5.2

VII. CONCLUSION & FUTURE SCOPE

On the comparison of both the cases means the apriori algorithm working on computer.arff file and the same algorithm working with the Weka tool , the result in both the cases it found that normally the apriori algorithm produce the best association rule for that dataset after performing the operation on it and the weka tool produse the 10 best association rules on that particular dataset for the same apriori algorithm and in the result of it weka produce 1 association rule same as the result of the apriori algorithm without using weka tool. Which showed that the algorithm produce the same result in both the cases.

So the observations and results are showing that this tool is capable to provide the proper exploration and analysis of the dataset and helpful to define the dataset and to take the decision in future aspects.

As the implementation of 'Apriori algorithm', it can be more compatible and purposeful in future, by implementation of the new association algorithms for some other new operations and analysis in this WEKA tool.

REFERENCES

[1] Rakesh Agrawal and Ramakrishnan Srikant, "Fast algorithms formining association rules in large databases", Proceedings of the 20th International Conference on Very Large Data Bases, VLDB, pages 487-499, Santiago, Chile, September 1994.
[2] http://www.cs.waikato.ac.nz/ml/index.html.
[3]http://eisc.univalle.edu.co/cursos/web/material/750061M/1/WekaManual.pdf
[4] Michael Hahsler and Sudheer Chelluboina, "Visualizing Association Rules in Hierarchical Groups", 42nd Symposium on the Interface: Statistical, Machine Learning, and Visualization Algorithms (Interface 2011).