Advances in Computer Science and Information Technology (ACSIT)
Print ISSN: 2393-9907; Online ISSN: 2393-9915; Volume 2, Number 3; January-March, 2015 pp. 262-266

http://www.krishisanskriti.org/acsit.html

A Study on Data Preprocessing, Classification and Clustering using WEKA Tool

Amit Gupta[1] and Naganna Chetty[2]

[1]Department of Computer Science and Engineering, Quantum School of Technology, Roorkee, Uttarakhand, India
[2]Department of Computer Science and Engineering, Mangalore Institute of Technology and Engineering, Moodabidri, Mangalore, Karnataka, India
E-mail: [1]amitgupta7920@gmail.com, [2]nsc.chetty@gmail.com

***Abstract*—*The term Data Mining is used to refer the process of analyzing large datasets and then extracting the knowledge from the data. In today's world data mining has become very essential in almost every area such as market segmentation, fraud detection, market based analysis, predictions of trends and behavior, discovery of previously unknown patterns etc. These can be achieved by designing and using various data mining software. This article introduces data preprocessing, classification and clustering in association with WEKA (Waikato Environment for Knowledge Analysis) data mining tool. The steps required to use WEKA tool for different data mining tasks through various algorithms are discussed. It also summarizes the applications of different data mining tasks.*

Keywords: *Data Mining, Classification, Clustering, preprocessing, WEKA tool, etc.*

1. INTRODUCTION

As we know, in general data mining refers to extraction of knowledge from huge amount of data. Practically, data mining (often refers to data or knowledge discovery) is defined as the process of analyzing the data according to various attributes and converting it into useful information – information that can be used for various purposes.

Data Mining is required due to wide availability of huge amount of data in electronic forms, and the requirement of converting such data into useful information so that it can be used in various applications. Data Mining is also referred as Knowledge Discovery in Databases (KDD) which consists of following steps and outlined in Fig. 1:

- Data Cleaning: to remove noise, irrelevant and inconsistent data
- Data Integration: combining multiple heterogeneous datasets into single set
- Data Selection: retrieval of relevant data from the database
- Data Transformation: conversion of data into a form, which is suitable for mining by performing summary or aggregation
- Data Mining: a very important step in which some algorithms and processes are applied for the extraction of patterns from the data
- Pattern Evaluation: used to identify the patterns that can represent some knowledge

Knowledge Presentation: in this step various representation techniques are used to present mined knowledge to the user.

Some common tasks in data mining are data preprocessing, pattern recognition, clustering and classification. In this article we discuss preprocessing, classification and clustering with WEKA data mining tool.

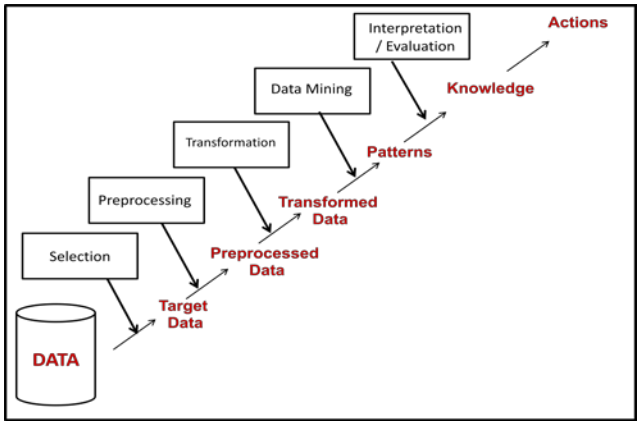

Fig. 1: The KDD Process

2. WEKA INTERFACE BACKGROUND

WEKA[9] stands for Waikato Environment for Knowledge Learning. It was developed by the University of Waikato, New Zealand. It is used to identify information from raw data gathered from various domains. WEKA supports many standard data mining tasks such as data preprocessing, classification, clustering, regression, visualization and feature

selection. It is an Open Source application that is freely available under the GNU public license agreement. It was originally written in C Language, but is converted to Java to make it compatible for every computing platform. WEKA provides access to SQL databases using Java Database Connectivity and can process the result returned by a database query.

JFreeChart [10] is an open-source framework for the programming language Java; it is an open source library available for Java that allows users to easily generate graphs and charts. It is particularly effective when a user needs to regenerate graphs that change on a frequent basis. JFreeChart supports pie charts (2D and 3D), bar charts, line charts, scatter plots, time series charts, and high-low-open- close charts.

The first interface that appears when WEKA started is given Fig. 2.

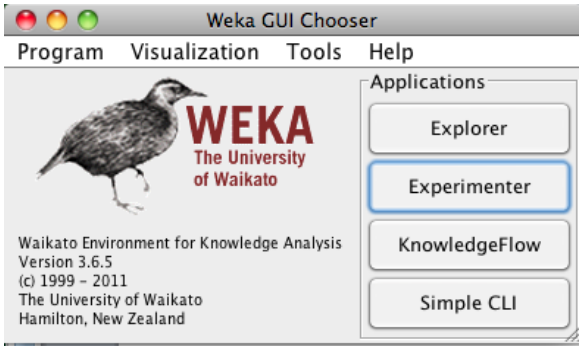

Fig. 2: The WEKA Interface

Explorer: It gives you an environment for exploring data. It supports Data Preprocessing, Attribute Selection, Learning and Visualization.

Experimenter: It's an environment for performing experiments and conducting statistical tests between machine learning algorithms.

Knowledge Flow: It is similar to Explorer but has a Drag-And-Drop interface. It gives a visual design of the KDD process.

Simple CLI: It provides a simple command line interface for running WEKA commands.

3. DATA PREPROCESSING AND ITS STEPS IN WEKA

Data Preprocessing is the technique that involves transforming raw data into a meaningful and useful form. Data Preprocessing is very important as the real world data is incomplete, inconsistent and noisy. Incomplete data means lacking some important attributes or contains only aggregate data. Incomplete data may come from not applicable data when collected, different consideration between the time when data was collected and analyzed. Noisy data (incorrect data) means that the data contains certain outliers and errors. Noisy data may come from faulty data collection instruments, data entry errors and errors in data transmission. Inconsistent data may arise due to different data sources and functions dependency violation.

Data Preprocessing is important because if there is no quality data, then there will be no quality mining results. Quality decisions must be based on quality data. Quality of result can be measured in terms of Accuracy, Completeness, Consistency, Timeliness, Accessibility and Interpretability.

Data Preprocessing can be done by Data Cleaning, Data Integration, Data Transformation, Data Reduction, Data discretization. Data Cleaning means filling of missing values, smooth noisy data to resolve inconsistency. Data integration is collection of multiple databases, data cubes, or files. Data transformation is the conversion of data into another form. Data reduction is reducing is reducing representation in volume but produces the similar analytical results. Data discretization is the part of data reduction but especially for numerical data.

Data Cleaning is one of the biggest problem in data warehousing. Missing values can be handled by simply ignoring the rows where the data is missing or filling the missing values manually. Noisy data can be handled by binning, clustering (detecting and removing outliers) and regression.

Data integration is the combination of data from multiple sources. Careful integration of data from multiple sources may help to reduce redundancies, inconsistencies and improves mining quality and speed.

Data Transformation can be done by smoothing (removing noise from data) and aggregation which can be achieved by summarization and data cube construction.

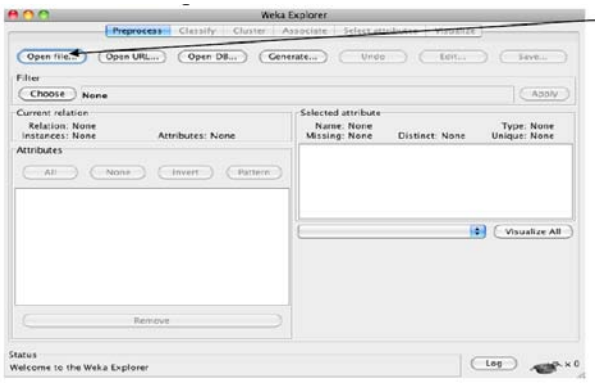

Fig. 3 GUI for Preprocessing in WEKA

The data file needs to be loaded first. The supported formats are ARFF, CSV, C4.5 and binary. Alternatively you can also import from URL or a SQL database. The GUI for preprocessing in WEKA is shown in Fig. 3.

After loading the data, preprocessing filters can be used for adding/removing attributes, sampling, classification etc.

4. CLASSIFICATION AND ITS STEPS IN WEKA

Classification is the process of finding a set of models that differentiate data classes and concepts. It is the technique used to predict group memberships for data instances. Classification [1] is the two step process: Model Construction that describes a set of predetermined classes. Each tuple is assumed to belong to a predefined class as determined by class label attribute, the set of tuples are used for model construction, called Training sets. The model is represented as Classification Rules, Decision Trees or Mathematical Formulae. Model usage that is used for classifying future data trends [2, 3] and unknown objects. It estimates the accuracy of the constructed model by using certain test cases. Test sets are always independent of the training sets.

There are four basic steps in WEKA for classification:

- Preparing the data
- Choose classify and apply algorithm
- Generate trees
- Analyze the result or output

In the following sub sections we discussed various Classification Algorithms [4, 5].

j48 (C4.5): J48 is an implementation of C4.5 [6] that builds decision trees from a set of training data in the same way as ID3, using the concept of Information Entropy. The training data is a set S = s1, s2... of already classified samples. Each sample si = x1, x2... is a vector where x1, x2… represent attributes or features of the sample. Decision tree are efficient to use and display good accuracy for large amount of data. At each node of the tree, C4.5 chooses one attribute of the data that most effectively splits its set of samples into subsets enriched in one class or the other.

Naive Bayes: A naive Bayes classifier assumes that the presence or absence of a particular feature is unrelated to the presence or absence of any other feature, given the class variable. Bayesian belief networks are graphical models, which unlikely naive Bayesian classifier; allow the representation of dependencies among subsets of attributes[7]. Bayesian belief networks can also be used for classification. A simplified assumption: attributes are conditionally independent:

$$P(C_j \mid V) \propto P(C_j) \prod_{i=1}^{n} P(V_i \mid C_j)$$

Where V is the data sample, V_i is the value of attribute i on the sample and C_j is the j^{th} class. It greatly reduces the computation cost, only count the class distribution.

k-nearest neighborhood: The k-NN algorithm for continuous-valued target functions Calculate the mean values of the k nearest neighbors Distance-weighted nearest neighbor algorithm Weight the contribution of each of the k neighbors according to their distance to the query point xqg giving greater weight to closer neighbors Similarly, for real-valued target functions. Robust to noisy data by averaging k-nearest neighbors.

The Euclidean distance between two points, $X=(x_1, x2,\ldots\ldots x_n)$ and $Y=(y_1, y2,\ldots\ldots y_n)$ is

$$d(x,y)= \sqrt{\sum_{i=1}^{n} (x_i - y)^2}$$

Neural Networks: Neural networks have emerged as an important tool for classification. The recent vast research activities in neural classification have established that neural networks are a promising alternative to various conventional classification methods. The advantage of neural networks lies in the following theoretical aspects. First, neural networks [8] are data driven self-adaptive methods in that they can adjust themselves to the data without any explicit specification of functional or distributional form for the underlying model.

Support Vector Machine: A new classification method for both linear and non linear data. It uses a nonlinear mapping to transform the original training data into a higher dimension. With the new dimension, it searches for the linear optimal separating hyper plane (i.e., “decision boundary”). With an appropriate nonlinear mapping to a sufficiently high dimension, data from two classes can always be separated by a hyper plane SVM finds this hyper plane using support vectors (“essential” training tuples) and margins (defined by the support vectors).

5. CLUSTERING AND ITS STEPS IN WEKA

Clustering is an automated process to group related records together on the basis of having similar values for the attributes. It is process of partitioning of set of datasets into set of meaningful sub classes called clusters. It helps user to understand natural grouping or structure in a dataset. Popular clustering techniques are:

K-Means Clustering: K-means is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori. The main idea is to

define k centroids, one for each cluster. These centroids should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early groupage is done. At this point we need to re-calculate k new centroids as barycenters of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop we may notice that the k centroids change their location step by step until no more changes are done. In other words centroids do not move any more.

Finally, this algorithm aims at minimizing an *objective function*, in this case a squared error function. The objective function is:

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2$$

where $\left\| x_i^{(j)} - c_j \right\|^2$ is a chosen distance measure between a data point $x_i^{(j)}$ and the cluster centre C_j, is an indicator of the distance of the *n* data points from their respective cluster centers.

Hierarchical Method: Hierarchical clustering arranges items in a hierarchy with a treelike structure based on the distance or similarity between them. The graphical representation of the resulting hierarchy is a tree-structured graph called a dendrogram. In Spotfire, hierarchical clustering and dendrograms are strongly connected to heat map visualizations. You can cluster both rows and columns in the heat map. Row dendrograms show the distance or similarity between rows, and which nodes each row belongs to as a result of clustering. You can perform hierarchical clustering in two different ways: by using the Hierarchical Clustering tool, or by performing hierarchical clustering on existing heat map visualization. If you use the Hierarchical clustering tool, a heat map with a dendrogram will be created. The algorithm used for hierarchical clustering in Spotfire is a hierarchical agglomerative method. For row clustering, the cluster analysis begins with each row placed in a separate cluster. Then the distance between all possible combinations of two rows is calculated using a selected distance measure. The two most similar clusters are then grouped together and form a new cluster. In subsequent steps, the distance between the new cluster and all remaining clusters is recalculated using a selected clustering method. The number of clusters is thereby reduced by one in each iteration step. Eventually, all rows are grouped into one large cluster. The order of the rows in a dendrogram is defined by the selected ordering weight. The cluster analysis works the same way for column clustering.

Agglomerative Approach: This is also known as bottom-up approach, in which each object forms a separate group. This technique merges the objects or groups that are close to each other until they form a single group.

In the agglomerative hierarchical approach, we start by defining each data point to be a cluster and combine existing clusters at each step. There are four different methods for doing this:

Single Linkage: In this step, we define the distance between two clusters to be the minimum distance between any single data point in the first cluster and any single data point in the second cluster. On the basis of this definition of distance between clusters, at each stage of the process we combine the two clusters that have the smallest single linkage distance.

Complete Linkage: In *complete linkage*, we define the distance between two clusters to be the maximum distance between any single data point in the first cluster and any single data point in the second cluster. On the basis of this definition of distance between clusters, at each stage of the process we combine the two clusters that have the smallest complete linkage distance.

Average Linkage: In *average linkage*, we define the distance between two clusters to be the average distance between data points in the first cluster and data points in the second cluster. On the basis of this definition of distance between clusters, at each stage of the process we combine the two clusters that have the smallest average linkage distance.

Centroid Method: In *centroid method*, the distance between two clusters is the distance between the two mean vectors of the clusters. At each stage of the process we combine the two clusters that have the smallest centroid distance.

Ward's Method: This method does not directly define a measure of distance between two points or clusters. It is an ANOVA based approach. At each stage, those two clusters marge, which provides the smallest increase in the combined error sum of squares from one-way univariate ANOVAs that can be done for each variable with groups defined by the clusters at that stage of the process

Divisive Approach: It is also known as top-down approach, in which all the objects lies under same cluster which recursively breaks ups into smaller clusters until there are multiple objects in a cluster.

Density-Based Method: Cluster analysis is a primary method for database mining. It is either used as a stand-alone tool to

get insight into the distribution of a data set, e.g. to focus further analysis and data processing, or as a preprocessing step for other algorithms operating on the detected clusters. Density-based approaches apply a local cluster criterion. Clusters are regarded as regions in the data space in which the objects are dense, and which are separated by regions of low object density (noise). These regions may have an arbitrary shape and the points inside a region may be arbitrarily distributed.

For other KDD applications, finding the outliers, i.e. the rare events, is more interesting and useful than finding the common cases, e.g. detecting criminal activities in E-commerce.

6. APPLICATIONS

Landuse Detection: DBSCAN has been applied to a 5-dimensional feature space created from several satellite images covering the area of California (5 different spectral channels: 1 visible, 2 reflected infrared, and 2 emitted (thermal) infrared). The images are taken from the raster data of the SEQUOIA 2000 Storage Benchmark. This kind of clustering application is one of the basic methods for automatic landuse detection from remote sensing data.

Potential Protein Docking Sites: The points on a protein surface (3d points) can be clustered with GDBSCAN to extract connected regions having a high degree of convexity or concavity. To find such regions is a subtask for the problem of protein-protein docking.

Influence Regions in GIS: GDBSCAN can be used to cluster 2*d* polygons from a geographic database, taking also into account the non-spatial attributes of the polygons. The found clusters, i.e. the so-called influence regions, are the input for a spatial trend detection algorithm.

Web-User Profiles : The goal of clustering Web-log sessions is to discover groups of access patterns which similar with respect to a suitable similarity measure. These access patterns characterize different user groups. This kind of information may, for example, be used to develop marketing strategies

7. CONCLUSION

WEKA is one of the common tool for data mining and it supports techniques for different data mining activities. It has been identified that preprocessing is essential to obtain better performance from the data mining system. It has been observed that it is convenient to mine data with WEKA data mining tool.

REFERENCES

[1] Serhat Özekes and A.Yilmaz Çamurcu,"Classification and Prediction in a Data Mining Application "Journal of Marmara for Pure and Applied Sciences, 18, pp. 159-174, 2002.

[2] Kaushik H and Raviya Biren Gajjar ,"Performance Evaluation of Different Data Mining Classification Algorithm Using WEKA", Indian Journal of Research(PARIPEX) vol. 2 issue 1, 2013.

[3] Wai Ho Au, Keith C., C. Chan and XinYao, "A Novel Evolutionary Data Mining Algorithm with Applications to Churn Prediction", IEEE Transactions on Evolutionary Computation, vol. 7, No. 6, PP. 532- 545, Dec 2003.

[4] Zhong, N.; Zhou, L.: "Methodologies for Knowledge Discovery and Data Mining", The Third Pacific-Asia Conference, Pakdd-99, Beijing, China, April 26-28, 1999; Proceedings, Springer Verlag.

[5] Raj Kumar, Dr. Rajesh Verma, "Classification Algorithms for Data Mining: A Survey" International Journal of Innovations in Engineering and Technology (IJIET).

[6] Fayyad, U. "Mining Databases: Towards Algorithms for Knowledge Discovery", IEEE Bulletin of the Technical Committee on Data Engineering, 21 (1) pp. 41-48, 1998.

[7] Qi Li and Donald W. Tufts:, "Principal Feature Classification" IEEE Transactions On Neural Networks, Vol. 8, No. 1, JANUARY 1997.

[8] Ling Liu: " From Data Privacy to Location Privacy: Models and Algorithms" September 23-28, 2007, Vienna, Austria.

[9] http://www.cs.waikato.ac.nz/ml/weka/downloading.html

[10] http://www.jfree.org/index.html.