Classification Algorithms and Comparison in Data Mining

Jigna Ashish Patel
Phd Student,
Nirma University, Ahmedabad

ABSTRACT
In present days, tons of data and information exist for each and everyone. Data can now be kept in many various kinds of databases as well as information repositories, besides being available online or in hard copy. With such big amount of data, a need for powerful techniques for better interpretation of these data that exceeds the human's ability for comprehension and making decision in a better way get into the picture. In order to get the best classification technique as well as tools required for handling with the classification task that helps in decision making, this survey has detailed a comparative study between a number of some data mining techniques and also tools required for its implementation. Results have shown that the performance of the tools for the classification task is overripe by the kind of dataset used and by the way the implementation of classification algorithms was done within the toolkits.

Introduction
Today’s databases and data repositories contain tons of data and information and so it becomes very tough, even impossible for a human being to evaluate them blue-collar for better decision making. So, they need some assistance or technique that can make work faster and efficient; therefore humans need techniques for data mining as well as its applications. [1]

Data mining is defined as the process of finding desired information from lots of amounts of data kept in databases and data warehouses as well as other information repositories. Also, a combination of techniques from multiple perspectives such as database and data warehousing technology, statistics, machine learning, high-performance computing, and pattern matching is also involved in data mining [2]. Marketing, business, science and engineering, economics, games and bioinformatics are also considered as the different fields of data mining.

As tons of information exist and from that, a particular part of it needs to be retrieved, some efficient methods should be used for its better operation.

1.1 Decision tree Algorithm
A decision tree is defined as a decision support system using a graph decisions of tree-like and their possible repercussions, including probability results, resource costs, and utility. A Decision Tree, also known as a classification tree, is used to discover a classification function that performs the operation of deducing the value of dependent attributes from the values of the independent attributes.

A decision tree is also defined as a flowchart-like structure in which a test on each node is represented by each node, the outcome is represented by each branch and a class label is represented by each leaf. The classification rule can be classified from the paths from root to leaf.

Some of the empirical application areas of decision trees are commonly operations research, specifically in decision analysis, to help recognize a strategy most suitable for reaching towards the goal.

1.1.1 Advantages and Disadvantages
Decision trees are taken as the most suitable approaches in information discovery as well as data mining. The technology of research big and complicated group of data with a view to find useful patterns are included in it [4]. Given approach is very essential because it enables modelling and
information retrieval from the group of information there to evaluate. All theoreticians and specialists are continually searching for methods to perform it in a more efficient way, economical and precise. In many fields apart from data mining like knowledge retrieval, machine learning, and pattern matching have application of decision tree. [7]

There are some benefits of decision tree algorithm as follow:

- Simple for understanding and relate properly to a set of production rules.
- Decision trees can be efficiently approached for real problems.
- No prior predictions about the behavior of the data are to be made.
- Efficient enough to create models with data containing numerical and also categorical values.

But it has some limitations compared to other algorithms that are as follow:

- Output attributes must be categorical, and more than one output attributes are not permitted.
- Not stable in that minor fluctuations in the training data can turn into various attribute selections at every choice node with in the tree. The effect can be worth-noticing as attribute choices adapts all descendent sub trees.
- Trees from numeric datasets can be more complex as attribute divided for numeric data are typically in a binary form.

1.1.2 Optimization

J48 is the java implementation of improved version of decision tree. [8] The improvements which are made are as follow:

- Managing continuous as well as discrete attributes
- Managing training data with not specified attribute values
- Modified of trees after its origination

With the advanced algorithm, quick and more efficient outcomes without the adaptation of the final decision can be achieved and the proposed algorithm makes the decision tree more specific and easy to understand. Also, improvisation in efficiency and categorization is achieved.[15]

1.2 K-Means Algorithm

K-means is a basic, simple partition clustering technique which operates to search a user-specified k number of clusters. Their centroids notifies these clusters that is typically the mean of the points in the cluster.

Two separate phases are involved in this algorithm: in the first phase, selection process of k centers at random is performed, where the value of k is constant from the start. During the next phase, Assignment of each data object to the nearest center is done. Euclidean distance is taken into consideration for determination of the distance from each data object to the cluster centers. After the inclusion of all the data objects in some clusters, recalculation operation is performed on the average of the clusters. This iterative process performs recursion until the criterion function reaches its minimum value. [12]

1.2.1 Algorithm steps

The steps involved in k-means algorithm are as follow:

- Select k data object from dataset S as initial cluster centers at random
- Repeat step 3 to step 5 till no new cluster centers are found.
- Measure the distance from each data object \(d_i\) (1\(\leq I\leq n\)) to all k cluster centers \(c_j\) (1\(\leq j\leq n\)) and assign data object \(d_i\) to the closest cluster.
- For each cluster j (1\(\leq j\leq k\)), perform recalculation of the cluster center. [13]

1.2.2 Variants of K-means algorithm

- Initialization of k
- modifying of center
- Migration of object from one cluster to another
1.2.3 Limitations
- Not applicable about categorical data unless mean is defined
- Specification of number of clusters in advance
- Not able to handle noisy data
- Not efficient enough to find clusters with non-convex shapes.

1.3 Tools:
An open-source development model usually means that the tool is a result of a community effort, not necessarily supported by a single institution but instead the result of contributions from an international and informal development team. This development style offers a means of incorporating the diverse experiences.

2
2.1 WEKA
WEKA (Waikato Environment for Knowledge Analysis) is a collection of machine learning algorithms for data mining tasks. WEKA is a Java based open source tool data mining tool which is a collection of many data mining and machine learning algorithms, including pre-processing on data, classification, clustering, and association rule extraction. WEKA provides three graphical user interfaces i.e. the Explorer for exploratory data analysis to support preprocessing, attribute selection, learning, visualization, the Experimenter that provides experimental environment for testing and evaluating machine learning algorithms, and the Knowledge Flow for new process model inspired interface for visual design of KDD process. A simple Command-line explorer which is a simple interface for typing commands is also provided by WEKA.

2.1.1 Pros and Cons
Advantages:
• No accessing cost
• Portability
• Detailed collection of data preprocessing and modeling technique
• Simple UI/UX
• Accessibility to SQL databases

Disadvantages:
• Improper and inadequate documentations and suffers from “Kitchen Sink Syndrome” where updating systems is done constantly.
• Connectivity issues to Excel spreadsheet and non-Java based databases.
• CSV reader not as robust as in Rapid Miner.
• Weaker in classical statistics.
• Does not have the feature to save parameters for scaling to use for future work.
• No automatic feature for Parameter optimization of machine learning/statistical methods.

2.2 Orange
Orange is considered to be a machine learning and component based software suite, expediting a visual programming front-end for explorative data conception and scrutiny, and Python trusses and libraries for scripting. [4] Inclusion of a set of components for data preprocessing, feature scoring and filtering, modeling, model evaluation, and exploration techniques is there. Implementation can be performed in C++ and Python. Its graphical user interface physiques upon the cross-platform context.

Orange also includes a set of graphical widgets that use methods from core library and Orange modules. It includes a set of components for data preprocessing, feature scoring and filtering, modeling, model evaluation, and exploration techniques. Through visual programming, widgets can be assembled together into an application by a visual programming tool called Orange Canvas.

2.2.1 Pros and Cons
Advantages:
• It is Python, NumPy, wrapped C, C++ and Qt based open source data mining package.
• Unswerving script for doing training, cross proof, algorithms appraisal and extrapolation.
• Simple script for data mining classification problems.
• Does not facilitate with optimum performance for connotation rules.

Disadvantages:
• Not super polished.
• Restricted list of machine learning algorithms.
• Does not handle machine learning uniformly between the different libraries.
• Recording abilities are restricted to spreading visual illustrations of data models.

Conclusion
Due to our survey on comparison among data mining classification’s algorithms) and analyzing of the time complexity of the mentioned algorithms we conclude that all decision Tree’s algorithms have less error rate and it is the easier algorithm as compared to KNN and Bayesian. Up to here and due to our survey based on the previously researches we extract the fact that among (Decision tree, KNN, K-means) algorithms in data mining, KNN is having lesser accuracy while Decision tree and Bayesian are equal. But if Decision tree algorithm has merged with genetic algorithm then in this way the accuracy of the Decision tree algorithm will improve and become more powerful and it will arise to be the best model approach among the other two algorithms. The efficiency of results using
KNN can be improvised by raising the number of data sets and for K-means algorithm classifier by increasing the attributes.

References