Volume 4, Issue 4, April 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Research on Data Mining Classification

Ritika
M.Tech Student
Department of CSE
Eternal University, Baru Sahib
Himachal Pradesh-173101, India

Abstract- The field Data mining has various important techniques and one of them is Classification which is receiving great attention recently in the database community. Classification technique can solve several problems in different fields like medicine, industry, business, science. Basically it involves finding rules that categorize the data into disjoint groups. There are several classification discovery models and these are: the decision tree, neural networks, genetic algorithms and some statistical models. This paper will discuss the details of classification which is described in the next sections.

Keywords: Data mining, Classification, Prediction, Training set, Prediction set, Classification Discovery Models

I. INTRODUCTION

In the present scenario there is enormous amount of data being collected and stored in databases everywhere across the world. It is not difficult to find the repositories with Terabytes of data in organizations and research fields. There is huge collection of data present and it is very difficult to extract important pieces of information out of it and without automatic extraction methods this information is practically impossible to mine. Year after year many algorithms were created to extract important information from large sets of data. There are different methodologies to approach this problem like classification rule, association rule, clustering, etc.

The input for the classification is the training data set, whose class labels are already known. Classification analyse the training data set and constructs a model based on the class label, and aims to assign class label to the future unlabelled records. Since the class field is known, this type of classification is known as supervised learning.

II. PROBLEM DESCRIPTION

Classification involves predicting an outcome based on a given input. In order to predict the outcome, the algorithm processes a training set containing a set of attributes and the respective outcome, normally known as prediction attribute. The algorithm discovers the relationships between the attributes that would make it possible to predict the outcome. After that the algorithm is given a new data set called prediction set, which contains the same set of attributes, except for the prediction attribute is not yet known. The algorithm analyses the input and generates a prediction. The accuracy of prediction defines how "good" the algorithm is. For example, in a blood donor's database the training set would have relevant donor's information recorded previously, where the prediction attribute is whether a person's response is positive or negative regarding donating the blood. Table 1 below illustrates the training and prediction sets.

Training set

Age	Blood Group	Test	Response
65	AB+	Not OK	-ve
19	O-	OK	+ve
28	B+	OK	Highly +ve

Prediction set

Age	Blood Group	Test	Response
50	AB+	Not OK	?
20	O-	OK	?
27	B+	OK	?

Table 1. Training and prediction sets for blood donor's database

Normally classification uses prediction rules to express knowledge. Prediction rules are expressed in the form of IF-THEN rules, where the antecedent (IF part) consists of a conjunction of conditions and the rule consequent (THEN part) predicts a certain predictions attribute value for an item that satisfies the antecedent. Using the example above, a rule predicting the third row in the training set may be represented as following:

IF (25<Age<45 AND Test=OK) OR (Age>=18 AND Blood Group= B+ AND Test=OK) THEN Response= Highly +ve.

In most cases the prediction rule is immensely larger than the example above. Conjunction has a nice property for classification; each condition separated by OR's defines smaller rules that captures relations between attributes. Satisfying any of these smaller rules means that the consequent is the prediction. Each smaller rule is formed with AND's which facilitates narrowing down relations between attributes. How well predictions are done is measured in percentage of predictions hit against the total number of predictions. A decent rule ought to have a hit rate greater than the occurrence of the prediction attribute. In other words, if the algorithm is trying to predict snowfall in Shimla and if it is snowing 80% of the time, the algorithm could easily have a hit rate of 80% by just predicting snowfall all the time. Therefore, 80% is the base prediction rate that any algorithm should achieve in this case. The optimal solution is a rule with 100% prediction hit rate, which is very hard, but not impossible, to achieve. Therefore, except for some very specific problems, classification by definition can only be solved by approximation algorithms.

III. CLASSIFICATION DISCOVERY MODELS

A) Decision Tree: Decision tree learning uses a decision tree as a predictive model which maps observations about an item to conclusions about the item's target value. It is one of the predictive modelling approaches used in statistics, data mining and machine learning. More descriptive names for such tree models are classification trees or regression trees. In these tree structures, leaves represent class labels and branches represent conjunctions of features that lead to those class labels. In decision analysis, a decision tree can be used to visually and explicitly represent decisions and decision making. In data mining, a decision tree describes data but not decisions; rather the resulting classification tree can be an input for decision making. Decision tree learning is a method commonly used in data mining. The goal is to create a model that predicts the value of a target variable based on several input variables. A decision tree is a simple representation for classifying examples. Decision tree learning is one of the most successful techniques for supervised classification learning.

Decision trees used in data mining are of two main types:

- **Classification tree** analysis is when the predicted outcome is the class to which the data belongs.
- **Regression tree** analysis is when the predicted outcome can be considered a real number (e.g. the price of a house, or a patient's length of stay in a hospital).

B) Neural Networks: Anyone can see that the human brain is superior to a digital computer at many tasks. A good example is the processing of visual information: a one-year-old baby is much better and faster at recognizing objects, faces, and so on than even the most advanced AI system running on the fastest supercomputer. The brain has many other features that would be desirable in artificial systems. Neural networks have the remarkable ability to derive meaning from complicated or imprecise data and can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques.

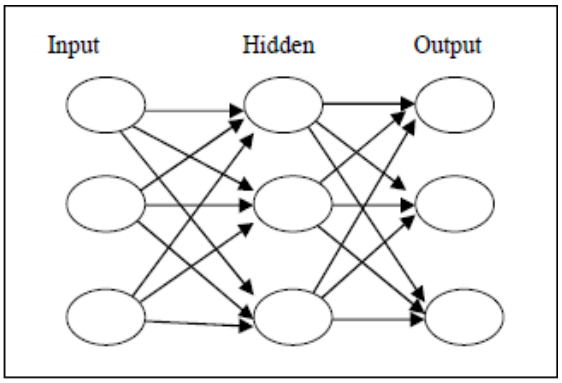

Fig. 1 Example of Neural Networks

Neural networks use a set of processing elements (or nodes) analogous to neurons in the brain. These processing elements are interconnected in a network that can then identify patterns in data once it is exposed to the data, i.e., the network learns from experience just as people do. This distinguishes neural networks from traditional computing programs that simply follow instructions in a fixed sequential order.

Neural networks essentially comprise three pieces: the architecture or model; the learning algorithm; and the activation functions. (Fausett (1994)) Neural networks are programmed or "trained to" . . . store, recognize, and associatively retrieve patterns or database entries; to solve combinatorial optimization problems; to filter noise from measurement data; to control ill defined problems. It is precisely these two abilities (pattern recognition and function estimation) which make artificial neural networks (ANN) so prevalent a utility in data mining. As data sets grow to massive sizes, the need for automated processing becomes clear. With their "model-free" estimators and their dual nature, neural networks serve data mining in a myriad of ways.

C) Genetic Programming: Genetic programming (GP) has been vastly used in research in the past 10 years to solve data mining classification problems. The reason genetic programming is so widely used is the fact that prediction rules are very naturally represented in GP. Additionally, GP has proven to produce good results with global search problems like classification. The search space for classification can be described as having several 'peaks', this causes local search

algorithms, such as simulated annealing, to perform badly. GP consists of stochastic search algorithms based on abstractions of the processes of Darwinian evolution. Each candidate solution is represented by an individual in GP. The solution is coded into chromosome like structures that can be mutated and/or combined with some other individual's chromosome. Each individual contains a fitness value, which measures the quality of the individual, in other words how close the candidate solution is from being optimal. Based on the fitness value, individuals are selected to mate. This process creates a new individual by combining two or more chromosomes, this process is called crossover. They are combined with each other in the hope that these new individuals will evolve and become better than their parents. Additionally to matting, chromosomes can be mutated at random. The running time of GPs is usually controlled by the user. There are many parameters used to determine when the algorithm should stop, and each data set can have very different settings. In all cases, the best individual is stored across generations and is returned when the algorithm stops.

The most commonly used parameter is number of generations. Another stop parameters used is minimum expected hit ratio, in which case the algorithm will run until a candidate solution has a hit ratio greater than expected. This however can cause the algorithm to run forever. Combinations of stop conditions can also be used to ensure stoppage.

D) Statistical Algorithms: ID3 AND C4.5

ID3 and C4.5 are algorithms introduced by Quinlan for inducing *Classification Models*, also called *Decision Trees*, from data. We are given a set of records. Each record has the same structure, consisting of a number of attribute/value pairs. One of these attributes represents the *category* of the record. The problem is to determine a decision tree that on the basis of answers to questions about the non-category attributes predicts correctly the value of the category attribute. Usually the category attribute takes only the values {*true, false*}, or {*success, failure*}, or something equivalent. In any case, one of its values will mean failure.

The basic ideas behind ID3 are that:

- In the decision tree each node corresponds to a non-categorical attribute and each arc to a possible value of that attribute. A leaf of the tree specifies the expected value of the categorical attribute for the records described by the path from the root to that leaf.
- In the decision tree at each node should be associated the non-categorical attribute which is *most informative* among the attributes not yet considered in the path from the root. [This establishes what a "Good" decision tree is.]
- *Entropy* is used to measure how informative is a node.

ID3 algorithm selects the best attribute based on the concept of entropy and information gain for developing the tree.

C4.5 algorithm acts similar to ID3 but improves a few of ID3 behaviours:

• A possibility to use continuous data.
• Using unknown (missing) values which have been marked by "?".
• Possibility to use attributes with different weights.
• Pruning the tree after being created.

E) Ant Colony:

Ant Colony algorithms were first introduced in the 1992 PhD thesis of Marco Dorigo. They offer a way of finding good paths within a graph, and they were inspired by the behaviour of ants in finding paths from the colony to food. When traveling from their colony to food sources, ants deposit chemicals called pheromones on the trails. The trail is used so that ants can find their way back to the colony. If other ants find this path they are likely to follow it. This will cause more pheromone to be deposited on the trail, which has the effect of reinforcing it. However, if a trail has not been used for a while, the pheromone starts to Evaporate. Short paths have the advantage of being marched over faster and more often, therefore, the pheromone density remains high. So a short path will have more ants traveling on it, increasing the pheromone density even more, and eventually all the ants will follow it. Ant algorithms mimic this behaviour in order to find optimal paths within a graph. Initially, all paths have a random small amount of pheromone deposited on it. An ant departs from the starting node and starts the process of visiting the other nodes in the graph. At each node, the ant decides which node it should visit next. The ant rates the attractiveness of traveling to each node in the graph. A nearby node with a large amount of pheromone deposited in its path will be very attractive. Also, a distant node with little amount of pheromone deposited in its path will be very unattractive. This attractiveness information will be use in order to compute the probability that a given route will be taken. Ant algorithms give good results relatively fast. They can be run continuously in order to adapt to changes in real-time: an advantage as compared to genetic algorithms. For instance, an obstacle such as a traffic jam would quickly lead the ants to discover a different good path. This can be of interest in applications such as network routing and urban transportation.

IV. CONCLUSION

With data mining we can find out various hidden patterns within large volume of data. These hidden patterns can be used to predict future behaviour. The availability of new data mining algorithms, however, should be met with caution. First of all, these techniques are only as good as the data that has been collected. Good data is the first requirement for good data exploration. If good data is already available, the next step is to choose the best technique to mine the data. However, there are trade-offs to consider when choosing the appropriate data mining technique to be used in a certain application. The most appropriate model is generally found by trial and error: trying different technologies and algorithms. Several times, the data analyst should compare or even combine available techniques in order to achieve the best possible outcomes/results.

REFERENCES

[1] Data Mining. March 2007. March 2007 <http://en.wikipedia.org/wiki/Datamining/>.
[2] Data Mining Techniques. 2006. March 2007<http://www.statsoft.com/textbook/stdatmin.html/>.
[3] Ant Colony Algorithm. January 2007. March 2007 http://www.egilh.com/blog/archive/2007/01/08/3322.aspx
[4] http://www.ijmlc.org/papers/184-C00015-001.pdf
[5] http://en.wikipedia.org/wiki/Decision_Tree
[6] http://www.cis.temple.edu/~ingargio/cis587/readings/id3-c45.html
[7] John Shafer, Rakesh Agarwal, and Manish Mehta, (1996) "SPRINT:A scalable parallel classifier for data mining", In Proc. Of the VLDB Conference, Bombay, India.
[8] Data Mining Techniques: Arjun K. Pujari
[9] Neural Networks based Data Mining and Knowledge Discovery in Inventory Applications: Kanti Bansal, Sanjeev Vadhavkar, Amar Gupta
[10] Artificial Neural Networks: Galgotia Publication
[11] Genetic Programming, John R. Koza, MIT Press, 1998.
[12] Genetic Programming An Introduction, W. Banzhaf, P. Nordin, R.E. Keller, F.D. Francone, MorganKaufmann Publishers, 1998.

