Global Journal of Computer Science and Technology
Software & Data Engineering
Volume 12 Issue 15 Version 1.0 Year 2012
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

Classification Rules and Genetic Algorithm in Data Mining

By Mr. Puneet Chadha & Dr. G.N. Singh

Panjab University, Chandigarh

Abstract - Databases today are ranging in size into the Tera Bytes. It is an information extraction activity whose goal is to discover hidden facts contained in databases. Typical applications include market segmentation, customer profiling, fraud detection, evaluation of retail promotions, and credit risk analysis. Major Data Mining Tasks and processes include Classification, Clustering, Associations, Visualization, Summarization, Deviation Detection, Estimation, and Link Analysis etc. There are different approaches and techniques used for also known as data mining models and algorithms. Data mining algorithms task is discovering knowledge from massive data sets. In this paper, we are focusing on Classification process in Data Mining.

GJCST-C Classification : *H.2.8*

Strictly as per the compliance and regulations of:

Classification Rules and Genetic Algorithm in Data Mining

Mr. Puneet Chadha [α] & Dr. G.N. Singh [σ]

I. Introduction

Databases today are ranging in size into the Tera Bytes. It is an information extraction activity whose goal is to discover hidden facts contained in databases. Typical applications include market segmentation, customer profiling, fraud detection, evaluation of retail promotions, and credit risk analysis. Major Data Mining Tasks and processes include Classification, Clustering, Associations, Visualization, Summarization, Deviation Detection, Estimation, and Link Analysis etc. There are different approaches and techniques used for also known as data mining models and algorithms. Data mining algorithms task is discovering knowledge from massive data sets. In this paper, we are focusing on Classification process in Data Mining.

The management and analysis of information and using existing data for correct prediction of state of nature for use in similar problems in the future has been an important and challenging research area for many years. Information can be analyzed in various ways. Classification of information is an important part of business decision making tasks. Many decision making tasks are instances of classification problem or can be formulated into a classification problem, viz., prediction and forecasting problems, diagnosis or pattern recognition. Classification of information can be done either by statistical method or data mining method.

II. Classification

Classification is a form of Data Analysis that can be used to construct a Model, which can be further used in future to predict the Class Label of new Datasets. Various Application of classification includes Fraud Detection, Target Marketing, Performance Prediction, Manufacturing and Medical Diagnosis.

Data Classification is a two step process

(i) The first step is a learning step. In this step a classification algorithm builds the Classifier by analyzing (or learning from)a training set made up of database tuples and their associated Class Labels. In this first step a Mapping Function Y=f(X) is learned that can predict the associated Class Label Y of a given tuple X. That mapping function or Classifier can be in the form of Classification Rules, Decision Trees or Mathematical Formulae.

(ii) Next Step of Classification, Accuracy of a Classifier is predicted. For this another set of tuples apart from training tuples are taken called as Test Sets. Then these set of tuples of test set are given as input to the Classifier.

The Accuracy of a Classifier on a given test set is the percentage of test set Tuples that are correctly classified by the Classifier.

a) Classification Methods in Data Mining

There are various Classification Methods as listed below

i. *Classification by Decision Tree Induction*

In this method Decision Tree is learned from Class-Labeled training tuples and then it is used for Classification. A Decision Tree is a flowchart–like tree structure, where each Internal Node (nonleaf node) denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node holds a class label.

While learning the Decision Tree or we can say during Tree Construction, attribute selection measures are used to select the attribute that best partitions the tuples into distinct classes. Once Decision Trees are built, Tree Pruning attempts to identify and remove branches that may reflect noise or outliers in the training data.

Learned Decision Trees can be used for Classification. Given a tuple X for which the associated Class Label is unknown, the attribute values of the tuple are tested against the decision tree. A path is traced from the root to a leaf node, which holds the class prediction for that tuple. Decision trees can be easily converted to Classification Rules.

ii. *Bayesian Classification*

Classifiers made using Bayesian Classification can predict the probability that a given tuple belongs to a particular Class.

Baye's Theorem: Using Baye's theorem we can predict Posterior Probability, P(H,X) from P(H),P(X|H) and P(X).Here X is a data tuple.

Baye's Theorem is

Author α : Assistant Professor, DAV college, Sector-10, affiliated to Panjab University, Chandigarh-160010.

Author σ : Head Department of Physics and Computer Science, Sudarshan Degree College, Lalgaon Distt. Rewa (M.P.) India.

$$P(H|X) = \frac{P(X|H)\,P(H)}{P(X)}$$

Where H ->Hypothesis such as that the data tuple X belongs to a specified Class C

P(H|X) ->Probability that hypothesis H exists for some given values of X's attribute.

P(X|H) ->Probability of X conditioned on H

P(H) ->Probability of H

P(X) ->Probability of X

Naive Bayesian Classification

Let d be a training set of tuples and $X=(x_1,x_2,x_3.....x_n)$ are the n attributes.

Let there be m classes $C_1,C_2.....C_m$.Naive Bayesian Classifier predicts that tuple X belongs to the class Ci if and only if

$$P(Ci|X) > P(Cj|X) \text{ for } 1<=j<=m,\ j \neq i$$

i.e X belongs to the Class having the Highest Posterior Probability.

Bayesian Belief Networks

The Naive Bayesian Classifier assumes Class Conditional Independence, but in practice dependencies can exist between Attributes (variables) or the tuple ie a particular categorization can depend on the values of two attributes.

Bayesian Belief Networks specify Joint Conditional Probability Distributions.

A Belief Network is defined by two components- A Directed Acyclic Graph and a Set of Conditional Probability Tables.

Each Node in the Graph correspond to actual attributes given in the data or to hidden variables. Each Arc represents a Probabilistic Dependence. Each variable is only Conditionally Dependent on its Immediate Parents. A Belief Network has one Conditional Probability Table (CPT) for each variable or attribute. In this the Conditional Probability for each known value of attribute is given for each possible combination of values of its Immediate Parents.

Trained Bayesian Belief Networks can be used for Classification. Various Algorithms for learning can be applied to the Network, rather than returning a Single Class Label.

The Classification Process can return a Probability Distribution that gives the Probability of each Class.

iii. *Rule Based Classification*

Rule Based Classifiers uses a set of IF-Then Rules for Classification.

IF Condition Then Conclusion.

The IF part is the Rule Antecedent or Precondition. Then Part is the Rule Consequent.

The Condition consists of one or more Attribute Tests that are logically ANDed. The Rule's Consequent contains a Class Prediction.

Let D be the Training data set. Let X be a tuple. If a Rule is satisfied by X, the rule is said to be triggered. Rule fires by returning the Class Prediction.

Rule Extraction from a Decision Tree

To Extract Rules from a Decision Tree, One Rule is created for each path from the root to a leaf node. Each Splitting Criterion along a given path is Logically ANDed to form the Rule Antecedent(IF part).The Leaf node holds the Class Prediction, forming the Rule Consequent(Then part).

Rule Induction using a Sequential Covering Algorithm

Here the Rules are Learned Sequentially, One at a time (for one Class at a time) directly from the Training Data (i.e without having to generate a Decision Tree first) using a Sequential Covering Algorithm.

iv. *Classification by Backpropagation*

Backpropagation is the most popular Neural Network Learning Algorithm. Neural Network is a set of connected input/output units, in which each connection has a weight associated with it. During the learning phase, the Network learns by adjusting the weights so as to be able to predict the Correct Class Label of the Input Tuples. Backpropagation performs on Multilayer Feed-Forward Neural Network. Several techniques have been developed for the Extraction of Rules from Trained Neural Networks. These factors contribute toward the usefulness of Neural Networks for Classification and Prediction in Data Mining.

v. *Support Vector Machines*

Support Vector Machines is a promising new method for the Classification of both Linear and Non Linear Data. Support Vector Machine is an algorithm that uses a Non Linear Mapping to transform the original training data into a higher dimension. Within this new dimension, it searches for Linear Optimal Separating Hyperplane (that is, a decision boundary separating the tuples of one class from another).The SVM finds this Hyperplane using Support Vectors (essential training tuples) and Margins (defined by the support vectors).

vi. *Associative Classification*

In Associative Classification, Association Rules are generated and analyzed for use in Classification. The general idea is that we can search for Strong Associations between Frequent Patterns (conjunctions of attribute-value pairs) and Class Labels. Because Association Rules explore highly confident Associations among Multiple Attributes, this approach may over-come some constraints introduced by Decision-Tree Induction which considers only one attribute at a time. In

particular three main methods are studied CBA, CMAR and CPAR.

vii. *Lazy Learners*

Lazy Learners do less work when a training tuple is presented and more work when making a Classification .When given a training tuple, a Lazy Learner simply stores it(or does a little minor processing) and waits until it is given a test tuple. Only when it sees the test tuple does it perform generalization in order to classify the tuple based on its similarity to the stored training tuples. Two examples of lazy learners are K-Nearest-Neighbour Classifiers and Case-Based Reasoning Classifiers.

III. Genetic Algorithm

A genetic algorithm (GA) is a search technique used in computing to find exact or approximate solution to optimization and search problems. Genetic algorithms are categories as global search heuristics.

Genetic algorithms are a probabilistic search and evolutionary optimization approach. Genetic algorithms are inspired by Darwin's theory about evolution. Solution to a problem solved by genetic algorithms is evolved.

Algorithm is started with a set of solutions (represented by chromosomes) called population. Solutions from one population are taken and used to form a new population. This is motivated by a hope, that the new population will be better than the old one. Solutions which are selected to form new solutions (offspring) are selected according to their fitness - the more suitable they are the more chances they have to reproduce.

This is repeated until some condition (for example number of populations or improvement of the best solution) is satisfied.

1. **[Start]** Generate random population of *n* chromosomes (suitable solutions for the problem)
2. **[Fitness]** Evaluate the fitness *f(x)* of each chromosome *x* in the population
3. **[New population]** Create a new population by repeating following steps until the new population is complete
    1. **[Selection]** Select two parent chromosomes from a population according to their fitness (the better fitness, the bigger chance to be selected)
    2. **[Crossover]** With a crossover probability cross over the parents to form a new offspring (children). If no crossover was performed, offspring is an exact copy of parents.
    3. **[Mutation]** With a mutation probability mutate new offspring at each locus (position in chromosome).
    4. **[Accepting]** Place new offspring in a new population
4. **[Replace]** Use new generated population for a further run of algorithm
5. **[Test]** If the end condition is satisfied, **stop**, and return the best solution in current population
6. **[Loop]** Go to step **2**

a) Genetic Algorithms and Classification

The construction of a classifier requires some parameters for each pair of attribute value where one attribute is the class attribute and another attribute is selected by the analyst. These parameters may be used as intermediate result for constructing the classifier. Yet, the class attribute and rest all attributes that analyst considers as relevant attributes must be the attributes of the tables that might be used for analysis in future. Hence, attribute values of class attribute are always frequent. When pre-computing the frequencies of pairs of frequent attribute values, the set of computed frequencies should also include the frequencies that a potential application needs as values of the class attribute and relevant attribute are typically frequent.

A framework for Genetic Algorithm to be implemented for Classification is

1. Start
2. Initialize the Population
3. Initialize the program size
4. Define the fitness f_i of an individual program corresponds to the number of hits and is evaluated by specific formula:
5. Run a tournament to compare four programs randomly out of the population of programs
6. Compare them and pick two winners and two losers based on fitness
7. a) Copy the two winners and replace the losers
    b) With Crossover frequency, crossover the copies of the winners
    c) With Mutation frequency, mutate the one of the programs resulting from performing step 7(a)
    d) With Mutation frequency, mutate the other of the programs resulting from performing step 7(a)
8. Repeat through step 5 till termination criteria are matched.

References Références Referencias

1. Piatetsky, Gregory. (2007). “Data mining and knowledge discovery 1996 to 2005: overcoming the hype and moving from “university” to “business” and “analytics”. *Data Min Knowl Disc (2007)* 15:99–105
2. Kriegel, Hans-Peter. Borgwardt, Karsten M. Kröger, Peer.*(et. al.)*(2007).“Future trends in data mining”. *Data Min Knowl Disc (2007)* 15:87–97
3. Luo, Qi. (2008). “Advancing Knowledge Discovery and Data Mining” *Knowledge Discovery and Data Mining, 2008.* WKDD 2008.

4. Weiss, Gary M. Zadrozny, Bianca. Saar-Tsechansky, Maytal. (2008). "Guest editorial: special issue on utility-based data mining". *Data Min Knowl Disc (2008)* 17:129–135.
5. Weber, Ben G. Mateas, Michael (2009). "A Data Mining Approach to Strategy Prediction" *978-1-4244-4815 2009 IEEE.*
6. Das, Sufal. Saha, Banani(2009). "Data Quality Mining using Genetic Algorithm". *International Journal of Computer Science and Security*, (IJCSS) Volume (3): Issue (2).
7. Kamble, Atul (2010). "Incremental Clustering in Data Mining using Genetic Algorithm". *International Journal of Computer Theory and Engineering*, Vol. 2, No. 3, June, 2010.
8. Kantarcıoglu, Murat. Xi, Bowei. Clifton, Chris. (2011). "Classifier evaluation and attribute selection against active adversaries" *Data Min Knowl Disc (2011)* 22:291–335.