Global Journal of Computer Science and Technology
Volume 12 Issue 2 Version 1.0 January 2012
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

Modified Tree Classification in Data Mining

By Raj Kumar, Dr. Anil Kumar Kapil, Anupam Bhatia
Assistant Professor, Kurukshetra University P.G. Regional Centre, Jind (India)

Abstract - Classification is a data mining technique used to predict group membership for data instances [1]. There are several conventional methods for classification in data mining like Decision Tree Induction, Bayesian Classification, Rule-Based Classification, Classification by Backpropagation and classification by Lazy Learners. In this paper we propose a new modified tree for classification in Data Mining. The proposed modified Tree is inherited from the concept of the decision tree and knapsack problem. A very high dimensional data may be handled with the proposed tree and optimized classes may be generated.

Keywords : *Classification, knapsack Problem, Modified Tree, KDD*

GJCST Classification: *E.1*

Strictly as per the compliance and regulations of:

Modified Tree Classification in Data Mining

Raj Kumar [α], Dr. Anil Kumar Kapil [Ω], Anupam Bhatia [β]

Abstract - Classification is a data mining technique used to predict group membership for data instances [1]. There are several conventional methods for classification in data mining like Decision Tree Induction, Bayesian Classification, Rule-Based Classification, Classification by Backpropagation and classification by Lazy Learners. In this paper we propose a new modified tree for classification in Data Mining. The proposed modified Tree is inherited from the concept of the decision tree and knapsack problem. A very high dimensional data may be handled with the proposed tree and optimized classes may be generated.

Keywords : Classification, knapsack Problem, Modified Tree, KDD

I. Introduction

Data mining or knowledge discovery is needed to make sense and use of data. Knowledge discovery in the data is the non-trivial process of identifying valid, novel, potentially useful and ultimately understandable patterns of data [2]. Data mining is the core step of knowledge discovery in database(KDD) and interdisciplinary field includes database management system, machine learning, statistics, neural network, fuzzy logic etc. Any of the technique may be integrated depending on the kind of data to be mined. The research in KDD is expected to generate a large variety of systems because diversity of disciplines to be contributed. Therefore a comprehensive classification system is required able to distinguish between the systems and identify the most required by the user. The major issue involved with the classification rule mining is to identify a dataset for a small number of rules to serve as classifier for predicting the class of any new instance. The classification algorithm should be accurate, simple and efficient. The existing classification algorithm assuming that the input data is drawn from a pre-defined distribution having stationary majors. Therefore these algorithms perform poorly when used to infer real world datasets.

In this paper, we propose a modified tree classification method which is result of knapsack implementation on Decision Tree Approach.

Author α : Assistant Professor, Deptt. of Comp Sc. and Engg., Jind Institue of Engineering & Technology, Jind (India)
E-mail : rajshira@gmail.com
Author Ω : Professor and HOD, Deptt. of CSE/IT, Haryana Institute of Engineering & Technology,Kaithal(india)
Author β : Assistant Professor, Kurukshetra University P.G. Regional Centre, Jind (India)

II. Classification Methods In Data Mining

Following classification methods are used in data mining:

(1) Decision Tree Induction: Decision tree is the learning of decision trees from a class labeled training tuples. A decision tree is a flow chart like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node holds a class label. The topmost node in a tree is the root node. The construction of decision tree classifier does not require any domain knowledge of parameter setting, and therefore is appropriate for exploratory knowledge discovery. Decision trees may handle high dimensional data. Their representation of acquired knowledge in tree from is intuitive and generally easy to understand by humans. The learning and classification steps of decision tree induction are simple and fast. In general, decision tree classifier has good accuracy. Decision tree induction algorithms have been used for classification in many applications areas, such as medicine, manufacturing and production, financial analysis, astronomy etc.[3]

(2) Rule-Based classification: In rule based classifiers, the learned model is represented as set of IF-THEN rules. We first examine how such rules are used for classification. Then we study the ways in which they may be generated, either from a decision tree of directly from the training data using a sequential covering algorithm. Rules are a good way of representing information or bits of knowledge. A rule-based classifier uses a set of IF-THEN rules for classification. An IF-THEN rule is and expression of the form:

IF *condition* THEN *conclusion*

The "IF" part of rule is known as the rule **antecedent or precondition**. The "THEN" part of the rule is known as the rule **consequent.** A rule R may be assessed by its coverage and accuracy. Given a tuple, X, from a class-labeled data set, D, let N_{COVERS} be the number of tuples covered by R; $N_{CORRECT}$ be the number of tuples correctly classified by R; and $|D|$ be the number of tuples in D. We may define **coverage** and **accuracy** of R as

$$\text{coverage}(R) = n_{covers} / |D|$$

$$\text{coverage}(R) = n_{corrects} / n_{covers} \quad [3]$$

(3) Classification by backpropagation: Backpropagation is a neural network learning algorithm. The field of neural networks was originally kindled by psychologists and neurobiologists who sought to develop and test computational analogues of neurons. A neural network is a set of connected input/output units in which each unit has a weight associated with it. Neural network learning is also referred to as connectionist learning due to the connections between units. Neural networks involve long training times and are therefore more suitable for applications where this is feasible. Backpropagation is the most popular neural network algorithm. Backpropagation learns by iteratively processing a data set of training tuples, comparing the network's prediction for each tuple with the actual known target value. The target value may be the known class label of the training tuple or continuous value. For each training tuple, the weights are modified so as to minimize the mean squared error between the network's prediction and the actual target value. These modifications are made in the "backwards" direction, that is, from the output layer, through each hidden layer down to the first hidden layer.[3]

(4) Lazy Learners: Decision tree induction, Bayesian classification, rule-based classification, classification by backpropagation, support vector machines are the example of the eager learner. Eager learners, when given a set of training tuples, will construct a generalization model before receiving new tuples to classify. Imagine a contrasting approach, in which the learner instead waits until the last minute before doing any model construction in order to classify a given test tuple. That is, when given a training tuple, a **lazy learner** simply stores it 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. Unlike eager learning methods, lazy learners do les work when a training tuple is presented and more work when making a classification or prediction. Because lazy learners store the training tuples of "instances", they are also referred to as **instance based learners.** When making a classification of prediction, lazy learners may be computationally expensive. They require efficient storage techniques and are well suited to implementation on parallel hardware. Lazy learners naturally support incrementing learning. K-nearest neighbor classifier and case-based reasoning classifiers are the example of the lazy learners. [3]

III. Motivation

While studying the Decision Tree Classification following shortcomings are observed[6].

- Decision trees are less appropriate for estimation tasks where the goal is to predict the value of a continuous attribute.
- Decision trees are prone to errors in classification problem with many classes and relatively with many class and relatively small number of training examples.
- The process of growing a decision tree is computationally expensive. At each node, each maydidate splitting field must be sorted before its best split may be found. In some algorithms, combinations of fields are used and a search must be made for optimal combining weights. Pruning algorithms may also be expensive since many maydidate sub-trees must be formed and compared.
- Most decision-tree algorithm only examine a single field at a time. This leads to rectangular classification boxes that my not correspond well with the actual distribution of records in the decision space.

Looking at these shortcomings, there is need to modify the decision tree classification into a new structure. In this paper in later section we propose a modified tree classification, the idea for which is to implement the concepts of Knapsack approach in Decision Tree Classification.

IV. Why Knapsack

Knapsack is well- known NP-hard problem. It has many applications like budgeting, network planning, network routing, and parallel scheduling etc. Knapsack problem may be of three types

0-1 knapsack problem: In the 0-1 knapsack problem the object is either completely selected or rejected in order to find the optimal solution. In the classical 0-1 knapsack problem only one knapsack is used and binary 1 represents that the object is selected and the binary 0 represents that the item is rejected.

Fractional knapsack: In the fractional knapsack problem the objects may be selected in fractions in order to maximize the profit or in other words we may say to find the optimal solution. For example in we have to select the objects like gold or diamond. Then these type of objects may be selected in fractions.

Multiple knapsack problem(MKP): Consider m containers (knapsacks) with capacities c_1, c_2,……,c_m and a set of n items, where each item has a weight w_1,w_2,……,w_n and a profit p_1,p_2,……p_n. Packing the items in the containers to maximize the total profit of the items, such that the sum of item weights in each container does not exceed the container's capacity, and each of item is assigned to at most one container, is the 0-1 multiple knapsack problem[4].

For example consider two knapsacks having capacities c_1=10, c_2=7. Suppose we have four items with weights 9,7,6,1 and profits 3,3,7,5. Now we have the problem that which item should be placed in which knapsack so that the profit may be maximized i.e. we

want to find out the optimal solution for this MKP. The optimal solution to this problem is move item 1 and 4 to the first knapsack, and item 3 to the second knapsack, it will give us a total profit of 15. So we may say that the MKP is the generalization of the classical 0-1 knapsack problem. Let the binary decision variable x_{ij} be one if item j is placed in container i, and 0 otherwise. The 0-1 MKP may be formulated as given below.

$$\text{maximize} \sum_{i=1}^{m} \sum_{j=1}^{n} p_j x_{ij}$$

$$\text{subject to:} \sum_{j=1}^{n} w_j x_{ij} \le c_i \qquad i=1,\ldots\ldots\ldots,m$$

$$\sum_{i=1}^{m} x_{ij} \le 1 \qquad j=1,\ldots\ldots\ldots,n$$

$$x_{ij} \in \{0,1\} \qquad \text{for all i, j}$$

So in this way 0-1 MKP may be represented as a classical 0-1 knap knapsack problem [5]. After looking the 0-1 multiple knapsack problem, we have observed that in the multiple knapsack problem more than one knapsacks are used and different objects may be placed in the different knapsacks in order to find the optimal solution. That is on the basis of certain attributes objects are assigned to a knapsack. Now in the classification in data mining, constraint set is defined for a class and a object satisfying the constraint set is assigned that class. Many classes may be there and each has its own constraint set. So in multiple knapsack problem multiple knapsacks (containers) are there and if we consider that objects in one knapsack have same attributes then we may consider a knapsack as class. So in this way both approaches have the probability of combination.

V. Modified Tree Classification

In this section, we propose a new tree classification method. The modified tree is inherited from the hybrid concept of the decision tree and the knapsack problem.

As shown in the figure1, Two knapsacks are defined.

1. The topmost node of the proposed tree (i.e. the root node) consists of all data sets.
2. On the left branch of the tree, a class is defined satisfying the first set of the attribute constraints as an external node. The left branch generate a new class.
3. Now the right subtree is explored recursively. The right branch is said to explore further recursively.
4. Each internal node represents the decision node, and each leaf node holds a class label.

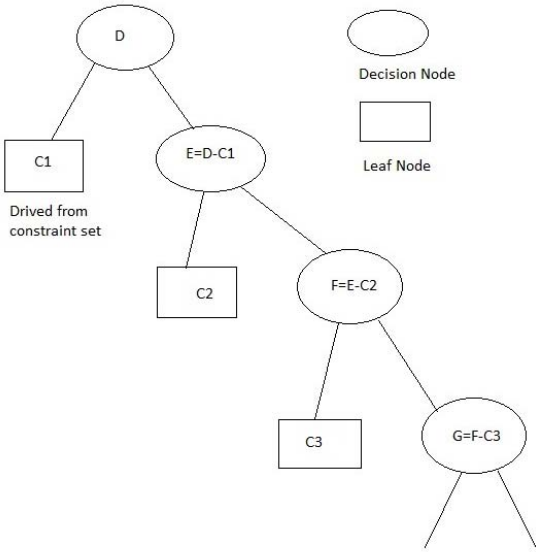

Figure1

With this approach the tree may handle and classify very high dimensional data effectively. However the complexity of the tree may be on higher side in comparison of the decision tree but the quality of the generated classes will definitely be higher. The leaf node of the proposed tree is represented by the rectangle and the internal nodes are represented by the oval.

In the figure1 D represents the all data set, on the left branch of the tree class C1 is derived from the constraint set defined for the class C1 (i.e. the objects in data set D satisfying the constraint set of C1 are placed in the class C1). So all the objects in the class C1 have the same attributes. Now the data set becomes E (E=E-C1). After this the subtree with the root node E is explored and on the left branch of the subtree class C2 is defined as the external node. Now data set becomes F(F=E-C2). Again the same method is applied to derive the class C3. In this way classes C1,C2,C3,…….and so on may be derived recursively in order to classify the data set.

This system has a number of advantages over various classified systems like decision tree induction, rule based classification and lazy learners. This technique may be used to solve a number of real world problems.

VI. Conclusion & Future Work

We proposed the modified tree for classification in data mining. The features of 0-1 Multiple Knapsack and decision tree are inherited. Greedy approach may be used to find the optimal solution of 0-1 multiple knapsack problem. Implementation of Artificial Intelligence (AI) techniques like Neural Network, Genetic

Algorithm, Simulation Annealing may be used for the above said classification method.

References Références Referencias

1. Thair Nu Phyu,"Survey of Classification Techniques in Data Mining"IMECS 2009
2. Fayyad, Piatetsky-Shapiro, Smyth and Uthurusamy,"Advances in Knowledge Discovery and Data Mining", (Chapter 1), AAAI/MIT Press 1996.
3. Jaiwei Han, Micheline Kamber,"Data MiPressning Concepts and Techniques, 2nd Edition", Morgan Kaufmann Publishers,2006 pp 289-301
4. G. Raidl, "The multiple container packing problem: A genetic approach with weighted coding," ACM SIGAPP Applied Computing Review, pp 22-32, 1999.
5. Alex S Fukunga, "A New Grouping Genetic algorithm for multiple Knapsack Problem", CEC 2008.
6. www.wikipedia.org

Global Journals Inc. (US) Guidelines Handbook 2012

www.GlobalJournals.org