Learning Decision Trees with Taxonomy of Propositionalized Attributes

Dae-Ki Kang and Kiwook Sohn
National Security Research Institute
161 Gajeong-Dong, Yuseong-Gu, Daejeon, South Korea
E-mail: dkkang@etri.re.kr kiwook@etri.re.kr

Abstract: We introduce Propositionalized Attribute Taxonomy guided Decision Tree Learner (PAT-DTL), an inductive learning algorithm that exploits a taxonomy of propositionalized attributes as prior knowledge to generate compact decision trees. Since taxonomies are unavailable in most domains, we also introduce Propositionalized Attribute Taxonomy Learner (PAT-Learner) that automatically constructs taxonomy from data. Our experimental results on UCI repository data sets show that the proposed algorithms can generate a decision tree that is generally more compact than and is often comparably accurate to those produced by standard decision tree learners.

1. Introduction

In typical learning scenario, instances to be classified are represented as ordered tuples of attribute values. Attribute values can be grouped together to reflect assumed or actual similarities among the values in a domain of interest or in the context of a specific application. Such a hierarchical grouping of values inside an attribute yields an attribute value taxonomy (AVT) [1]. Abstraction of similar concepts by means of attribute value taxonomy (AVT) has been shown to be useful in generating concise and accurate classifiers[2,3]. Those researches are only interested in abstraction of values inside a certain attribute. A set of values from different attributes may be grouped together to reflect a useful concept. To facilitate this grouping, an attribute will be propositionalized into a set of Boolean attributes, each of which corresponds a value of the original attribute. The hierarchical abstraction of the propositionalized attributes produces propositionalized attribute taxonomy (PAT). However, there has been no research on the analysis on the correlation of values from different attributes in a data set.

Against this background, we introduce propositionalized attribute taxonomy guided decision tree learner (PAT-DTL). PAT-DTL is an extension of C4.5 decision tree learning algorithm[4] to utilize PAT. PAT-DTL uses both top-down and bottom-up search techniques over the PAT to find a useful abstraction for classification task. To generate PAT automatically, we devise a propositionalized attribute taxonomy learner (PAT-Learner) that automatically generates PAT from data by clustering propositionalized attributes based on their class conditional distribution. To evaluate our algorithms, we conducted experiments with 37 data sets from UCI machine learning repository[5]. We used PAT-Learner to generate PAT from the training data. The generated PAT was provided to PAT-DTL to learn concise decision tree classifiers that used abstract values of PAT.

2. Propositionalized Attribute Taxonomy guided Decision Tree Learner (PAT-DTL)

2.1 Propositionalized Attribute Taxonomy

Propositionalization is a function that, for each value associated with each attribute of a data set, constructs a new Boolean attribute. Propositionalized attribute value associated with the new Boolean attribute is a Boolean value in \{True, False\}. An attribute of a propositionalized instance has a value True if and only if the original instance has the corresponding attribute value. A propositionalized attribute taxonomy (PAT) is defined over the Boolean attributes propositionalized from the original attribute. After Haussler[6], we define a cut through a propositionalized taxonomy as a subset of nodes in the taxonomy satisfying the following two properties:

- For any leaf \(x\), either \(x\) is in the cut or \(x\) is a descendant of a node in the cut.
- For any two nodes \(x, y\) in the cut, \(x\) is neither a descendant nor an ancestor of \(y\).

A cut of a propositionalized taxonomy induces a partition of propositionalized attributes. A cut is refined if it is obtained by replacing at least one node by its descendants. Conversely, a cut is abstracted if all the nodes in the cut are replaced by their parent. We can see that any choice of a cut over a propositionalized taxonomy defines a propositionalized instance space.

2.2 PAT-DTL Algorithm

The problem of learning classifiers from a propositionalized attribute taxonomy and propositionalized data is an extension of the problem of learning classifiers from the data. Original data set is a collection of labeled instances and a classifier is a hypothesis in the form of function where its domain is a set of instances and its range is a set of class labels. A hypothesis space is a set of hypotheses that can be represented in a hypothesis language or by a parameterized family of functions. Then, the task of learning classifiers from the data set is induce a hypothesis that satisfies given criteria.

Similarly, the problem of learning classifiers from propositionalized attribute taxonomy and propositionalized data can be described as follows: Given a propositionalized attribute taxonomy and a propositionalized data set, the aim is induce a classifier where its domain is a set of instances derived from a propositionalized instance space such that the corresponding cut maximizes given criteria. PAT-DTL is a top-down (or bottom-up) multi-level PAT guided hill-climbing search algorithm in the hypothesis space, and it is composed of two major components: (a) computation of the counts based on the given PAT, (b) and construction of the decision tree based on the counts.
The search for the locally optimal cut of PAT-DTL over the propositionalized attribute taxonomy is performed based on the accuracy of the decision trees generated from the training data as a model evaluation criterion. The accuracy for model selection is measured by five-fold stratified cross validation scheme to avoid over-fitting, and the decision trees are constructed by C4.5 decision tree learning algorithm [4]. The algorithm terminates when none of candidate refinements of the classifier yield statistically significant improvement in the given model evaluation criterion. There are two different ways for hill-climbing search of the locally optimal cut: top-down and bottom-up. PAT-DTL uses cut refinement for top-down search and cut abstraction for bottom-up search.

3. Learning Propositionalized Attribute Taxonomy

3.1 Problem Definition

The problem of learning propositionalized attribute taxonomy (PAT) from data can be stated as follows: given a propositionalized data set and a measure of dissimilarity (or equivalently similarity) between any pair of Boolean attributes, output a PAT such that the taxonomy corresponds to a hierarchical grouping of propositionalized Boolean attributes based on the specified similarity measure.

3.2 Algorithm

We use hierarchical agglomerative clustering (HAC) of Boolean attributes based on the distribution of class labels that co-occur with the attributes when they are True. The basic idea is to construct a taxonomy by starting with the primitive attributes as the leaves of the taxonomy and recursively add nodes to the taxonomy one at a time by merging two existing nodes. Let $\text{DM}(P(x)||Q(x))$ denote a measure of pairwise divergence between two probability distributions $P$ and $Q$ of the random variable $x$. We use a pairwise measure of divergence between the distributions of the class labels associated with the corresponding Boolean attributes as a measure of dissimilarity between them. The lower the divergence between the class distribution between two attributes, the greater is their similarity. The choice of this measure of dissimilarity is motivated by the intended use of propositionalized attribute taxonomy for PAT-DTL algorithm to generate accurate decision trees. If two attributes are indistinguishable from each other with respect to their class distribution, they will provide statistically similar information for classification of instance.

The algorithm maintains a cut through the taxonomy, updating the cut as new nodes are added to the taxonomy. At each step, the two words to be grouped together to obtain an abstract word to be added to the taxonomy are selected from the cut based on the divergence between the class distributions associated with the corresponding words. That is, a pair of words in the cut is merged if they have more similar class distributions than any other pair of words in the cut. This process terminates when the cut contains a single word which corresponds to the root of the taxonomy.

3. Experiments

We conducted experiments on 37 data sets from UCI Machine Learning Repository[5]. We test four different settings: C4.5[4] decision tree learner on the original attributes, C4.5 decision tree learner on propositionalized attributes, PAT-DTL with abstraction and PAT-DTL with refinement. 10-fold cross-validation is used for evaluation. In each case, a taxonomy is generated using PAT-Learner and a decision tree is constructed using PAT-DTL on the resulting PAT and data. The results of experiments described in this section reflect that none of the algorithms shows the highest accuracy over most data sets. However, as for the tree size, PAT-DTL coupled with PAT-Learner usually generates more concise decision trees (35 of 37 data sets). And, PAT-DTL often produces comparably accurate classifiers to those by C4.5 decision tree inducer (17 of 37 data sets). PAT-DTL with refinement generally yields most concise trees, however, for nine of 37 data sets tested, it yields decision trees that are most accurate too.

4. Summary

We have presented Propositionalized Attribute Taxonomy guided Decision Tree Learner (PAT-DTL) that exploits a taxonomy of propositionalized attributes to generate compact classifiers, and Propositionalized Attribute Taxonomy Learner (PAT-Learner) that automatically constructs taxonomy from data. PAT-DTL uses both top-down and bottom-up search based learning algorithm that generates decision tree classifiers from data and propositionalized attribute taxonomy. PAT-Learner is an hierarchical agglomerative clustering algorithm that clusters attributes based on the distribution of class labels that co-occur with them. Experimental results on UCI repository data sets show that the proposed algorithms can generate decision tree classifiers that are more compact and often comparably accurate to those produced by standard decision tree learner.

References