Learning Naïve Bayes Classifiers From Attribute Value Taxonomies and Partially Specified Data

Jun Zhang and Vasant Honavar

Technical Report ISU-CS-TR-04-03

Department of Computer Science

Iowa State University Ames, IA 50011

February 2004

Learning Naïve Bayes Classifiers from Attribute Value Taxonomies and Partially Specified Data

Jun Zhang JZHANG@CS.IASTATE.EDU
Vasant Honavar HONAVAR@CS.IASTATE.EDU
Artificial Intelligence Research Laboratory, Department of Computer Science, Iowa State University
Ames, IA 50011 USA

Abstract

Partially specified data are commonplace in many practical applications of machine learning where different instances are described at different levels of precision relative to an attribute value taxonomy (AVT). This paper describes AVT-NBL – a variant of the Naïve Bayes Learning algorithm that effectively exploits user-supplied attribute value taxonomies to construct compact and accurate Naïve Bayes classifiers from partially specified data. Our experiments with several data sets and AVTs show that AVT-NBL yields classifiers that are substantially more accurate and more compact than those obtained using the standard Naïve Bayes learner.

1. Introduction

In many pattern classification tasks, it is often the case that the instances to be classified are specified at different levels of precision [Zhang and Honavar, 2003]. That is, the value of a particular attribute, or the class label associated with an instance, or both are specified at different levels of precision in different instances, leading to *partially specified instances.* To illustrate this phenomenon, an attribute value taxonomy (AVT) for the "color" attribute in which color takes on several values – *Blue*, *Red*, etc. is shown in Figure 1. Now suppose that *Blue* objects can be further specified in terms of the precise shade of blue such as *Sky Blue*, *Light Blue*, *Dark Blue* and *Navy Blue.* In this case, in one instance, the color of a particular object may be described as *navy blue*, whereas in another instance, it may be specified simply as *Blue* without specifying the precise shade of blue.

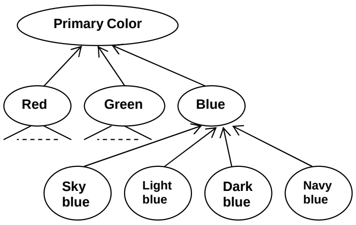

Figure 1. A Value Taxonomy for Color Attribute

Algorithms for learning from AVT and partially specified data are of significant practical interest for several reasons:

a. Partially specified data are quite common in many application domains including medical diagnosis, scientific discovery, electronic commerce, and security informatics. For example, in a medical diagnosis task, different cases may be described in terms of symptoms or results of diagnostic tests at different levels of precision e.g., a patient may be described as having cardiac arrhythmia without specifying the precise type of arrhythmia.
b. Partially specified data are unavoidable in knowledge acquisition scenarios which call for integration of information from semantically heterogeneous, information sources [Reinoso-Castillo et al., 2003; Caragea et al., 2004]. Semantic differences between information sources arise as a direct consequence of differences in ontological commitments (i.e., assumptions about the objects and the properties of objects in the domain of interest) [Berners-Lee et al., 2001]. Taxonomies and part-whole hierarchies are among the most common and useful types of ontologies. Increasing need for data sharing between autonomous organizations and groups have led to major efforts aimed at construct ion of taxonomies (e.g., AVT). Examples include ontologies for describing many aspects of macromolecular sequence, structure, and function e.g., gene ontology (www.geneontology.org) [Ashburner et al., 2000], and ontology for intrusion detection [Undercoffer et al. 2003].
c. An important goal of machine learning is to discover comprehensible, yet accurate and robust classifiers [Pazzani et al., 1997]. The availability of AVT presents the opportunity to learn classification rules that are expressed in terms of *abstract* attribute values (e.g., color=*Blue* instead of color=*Navy Blue*) leading to simpler, easier-to-comprehend rules that are expressed in terms of familiar hierarchically related concepts. Kohavi and Provost [2001] have

noted the need to be able to incorporate hierarchically structured background knowledge e.g., hierarchies over data attributes in electronic commerce applications of data mining. Similar considerations arise in applications in diagnosis and scientific discovery.

d. When training data are limited, there is a risk of generating classifiers that over fit the training data. A common approach used by statisticians when estimating from small samples involves *shrinkage* [McCallum et al, 1998] or grouping attribute values (or more commonly class labels) into bins when there are too few instances that match any specific attribute value or class label to estimate the relevant statistics with adequate confidence. Learning algorithms that exploit AVT can potentially perform *shrinkage* automatically thereby yielding robust classifiers. In other words, exploiting information provided by an AVT can be an effective approach to performing regularization to minimize over-fitting [Zhang and Honavar, 2003].

e. In many applications, there is a need to explore data from multiple points of view*,* or working assumptions on the part of the learner. The choice of the working assumptions (e.g., in the form of an AVT) in learning from data is analogous to the choice of axioms in mathematics. An AVT that captures the relevant relationships among attribute values can result in the generation of simple and accurate classifiers from data, just as an appropriate choice of axioms in a mathematical domain can simplify proofs of theorems. Thus, the simplicity and predictive accuracy of the learned classifiers based on alternative choices of AVT can be used to evaluate the utility of the corresponding AVT in specific contexts.

Against this background, this paper introduces AVT-NBL, an AVT-based generalization of the standard algorithm for learning Naïve Bayes classifiers from data.

2. Learning Classifiers from AVT and Partially Specified Data

In what follows, we define AVT, introduce the notions of a partially missing value (relative to an AVT), and a partially specified instance (relative to the AVTs associated with the attributes used to describe instances) [Zhang and Honavar, 2003]. An Attribute Value Taxonomy (AVT) associated with an attribute α, AVT(α) is a tree rooted at α. The set of leaves of the tree, *Leaves*(AVT(α)), corresponds to the set of possible *primitive values* of A. The internal nodes of the tree correspond to *abstract values* of attribute α. The *arcs* of the tree correspond to ISA *relationships* between attribute values that appear in adjacent levels in the tree. Let *Nodes*(AVT(α)) denote the set of nodes of the AVT associated with attribute α. Figure 1 shows an example of an AVT for the *color* attribute. The set of abstract values at any given level in the tree AVT(α) form a partition of the set of values at the next level (and hence, the set of primitive values of α). For example, in Figure 1, the nodes at level 1, i.e., *Red*, *Green*, *Blue*, define a partition of attribute values that correspond to nodes at level 2 (and hence, a partition of all primitive values of the 'color' attribute). After Haussler [1988], we define a cut Z of an AVT(α) is a subset of nodes in AVT(α) satisfying the following two properties: (1) For any leaf $l \in$ *Leaves*(AVT(α)), either $l \in Z$ or l is a descendent of a node $n \in Z$; and (2). For any two nodes $f, g \in Z$, f is neither a descendent nor an ancestor of g. Cuts through AVT(α) correspond to a partition of *Leaves*(AVT(α)). Thus, the *cut* corresponding to {*Red, Green, Sky Blue, Dark Blue, Navy Blue, Light Blue*} defines a partition over the primitive values of the 'color' attribute.

When each attribute has a single AVT, we will use $T=\{T_1, T_2, ..., T_N\}$ (where $T_i=$ AVT (A_i)) to represent the set of the corresponding AVTs. Let $Root(T_i)$ stand for the root of the AVT T_i. To make the notion of partially specified instances more precise, we define several operations on an AVT taxonomy T_i associated with an attribute A_i.

(1) $depth(T_i , v(A_i))$ returns the length of the path from root to an attribute value $v(A_i)$ in the taxonomy;

(2) $leaf(T_i , v(A_i))$ return a Boolean value indicating if $v(A_i)$ is a leaf node in T_i =AVT(A_i), that is if $v(A_i) \in Leaves(T_i)$.

With respect to an AVT, a (completely) missing value of an attribute α corresponds to the root of AVT(α). We say that an attribute α is fully specified in an instance with respect to AVT(α) when the value of attribute α is a primitive value of α. We say that the value of an attribute α is partially specified (or equivalently, partially missing) when its value is not one of the primitive values of α. Thus, we can have instances specified at different levels of precision resulting in *partially specified instances.* An instance I_j is expressed as a tuple $I_j=(v_1^{(j)}, v_2^{(j)},\dots, v_n^{(j)})$ where each attribute A_i has a corresponding AVT T_i. I_j is:

- a completely specified instance if $\forall i \ v_i^{(j)} \in Leaves(T_i)$
- a partially specified instance when one or more of its attribute values are not primitive: $\exists v_i^{(j)} \in I_j$, $depth(T_i, v_i^{(j)}) \geq 0 \wedge \neg leaf(T_i, v_i^{(j)})$

Thus, a partially specified instance is an instance in which at least one of the attributes is partially specified. For example, consider a set of objects described in terms of the attributes 'color' and 'shape'. The AVT for color is shown in Figure 1. Suppose *Triangle*, *Polygon* are *primitive values* of the 'shape' attribute (The AVT for shape is not shown). (*Light Blue, Triangle*), is an example of a fully specified instance. Some examples of partially specified instances are (*Blue, Polygon*), (*Dark Blue, Polygon*), and (*Blue, Square*).

The problem of learning classifiers from AVT and partially specified data can be stated as follows: Given a user-supplied set of AVTs and a data set of (possibly) partially specified labeled instances, construct a classifier *h* from a suitable hypothesis class *H* for assigning partially specified instances to one of several mutually exclusive classes. For any hypothesis class H_F (e.g., decision trees, Naïve Bayes classifiers) defined over an instance space corresponding to fully specified instances described by a set of attributes, and an AVT, we can define a hypothesis class H_P over an instance space of partially specified instances induced by the AVT.

3. Approaches to Learning Classifiers from Partially Specified Data

We can envision three approaches to learning from Partially Specified Data:

Approaches that Treat Partially Specified Attribute Values as if they were Totally Missing: Each partially specified (and hence partially missing) attribute value is treated as if it were (totally) missing, and the resulting data set with missing attribute values is handled using standard approaches for dealing with missing attribute values in learning classifiers from an otherwise fully specified data set in which some attribute values are missing in some of the instances values. A main advantage of this approach is that it requires no modification to the learning algorithm. All that is needed is a simple preprocessing step in which all partially specified attribute values are turned into missing attribute values.

AVT-Based Propositionalization Methods: The data set is represented using a set of Boolean attributes obtained from AVT T_i of attribute A_i by associating a Boolean attribute with each node (except the root) in T_i. Thus, each instance in the original data set defined using *N* attributes is turned into a Boolean instance specified using *L* Boolean attributes where

$$L = \left(\sum_{i=1}^{N} \left| Nodes(T_i) \right| \right).$$

In the case of the color taxonomy shown in Figure 1, this would result in binary features that correspond to the propositions such as (color=*Red*), (color=*Blue*), (color=*Green*), … (color=*Sky Blue*) … (Color=*Navy Blue*). Based on the specified value of an attribute in an instance e.g., (color = *Sky Blue*), the values of its ancestors in the AVT (e.g., color=*Blue*) are set to True because the AVT asserts that *Sky Blue* objects are also *Blue* objects. But the Boolean attributes that correspond to descendents of the specified attribute value are treated as unknown. For example, when the value of the color attribute is partially specified in an instance, e.g. (Color=*Blue*), the corresponding Boolean attribute is set to True, but the Boolean attributes that correspond to the descendents of *Blue* in the color taxonomy are treated as *missing*. The resulting data with some missing attribute values can be handled using standard approaches to dealing with missing attribute values.

Note that the Boolean features created by the propositionalization technique described above are not independent given the class. A Boolean attribute that corresponds to any node in an AVT is necessarily correlated with Boolean attributes that correspond to its descendents as well as its ancestors in the tree. For example, the Boolean attribute (color=*Blue*) is correlated with (color=*Sky Blue*). (Indeed, it is this correlation that enables us to exploit the information provided by AVT in learning from partially specified data). Thus, a Naïve Bayes classifier that would be optimal in the Maximal a Posteriori sense [Langley et al., 1992; Mitchell, 1997] when the original attributes Color and Shape are independent given class would no longer be optimal when the new set of Boolean attributes are used because of dependencies among the Boolean attributes derived from an AVT**.**

A main advantage of the AVT-based propositionalization methods is that they require no modification to the learning algorithm. However it does require preprocessing of partially specified data using the information supplied by an AVT. The number of attributes in the transformed data set is substantially larger than the number of attributes in the original data set. More importantly, the statistical dependence among the Boolean attributes in the propositionalized representation of the original data set can degrade the performance of classifiers e.g., Naïve Bayes that rely on independence of attributes given class. Hence, it is of interest to explore principled approaches to exploiting the information provided by AVT in learning classifiers from partially specified data.

AVT Guided Variants of Standard Learning algorithms: We can extend standard learning algorithms in principled ways so as to exploit the information provided by AVT. AVT-DTL [Zhang & Honavar, 2003] which extends the standard decision tree learning algorithm and the AVT-NBL algorithm described in this paper which extends the standard algorithm for learning Naïve Bayes classifiers are examples of this class of algorithms.

It is interesting to explore the performance of alternative approaches to learning classifiers from partially specified data.

4. AVT-Based Naïve Bayes Learner (AVT-NBL)

4.1 Naïve Bayes Learner (NBL)

Let A_1 ... A_N be an ordered set of attributes and $C = \{c_1, c_2, \ldots, c_M\}$ a finite set of mutually disjoint classes. Suppose each attribute A_i takes a value from a finite set of values $V(A_i)$. An instance $\mathbf{X}_p$ to be classified is represented as a tuple of attribute values $(a_{1p}, a_{2p}, \ldots, a_{Np})$ where each

$a_{ip} \in V(A_i)$. The Bayesian approach to classifying $X_p = (a_{1p}, a_{2p}, \ldots, a_{Np})$ is to assign it the most probable class $c_{MAP}(\mathbf{X}_p)$ given the attribute values $\mathbf{X}_p = (a_{1p}, a_{2p}, \ldots, a_{Np})$. Naïve Bayes classifier operates under the assumption that each attribute is independent of others given the class. Hence, we have:

$$c_{MAP}(X_p) = \arg\max_{c_j \in C} P(a_{1p}, a_{2p}, \ldots, a_{Np} \mid c_j) p(c_j)$$
$$= \arg\max_{c_j \in C} p(c_j) \prod_i P(a_{ip} \mid c_j)$$

The standard algorithm (NBL) for learning a Naïve Bayes classifier simply estimates a class conditional probability table for each attribute from a data set D of training examples. The class conditional probability table for attribute A_i has $|V(A_i)||C|$ entries. The probabilities are typically estimated using a Bayesian approach [Mitchell, 1997].

4.2 AVT-Guided Naïve Bayes Learner (AVT-NBL)

Given a user-supplied ordered set of AVTs $T_1 \ldots T_N$ corresponding to the attributes $A_1 \ldots A_N$ and a data set $D = \{(\mathbf{X}_p, c_p)\}$ of labeled examples of the form $(\mathbf{X}_p, c_p)$ where $\mathbf{X}_p$ is a partially specified instance and c_p is the corresponding class label, the task of AVT-NBL is to construct a Naïve Bayes classifier for assigning a partially specified instance $\mathbf{X}_p$ to its most probable class $c_{MAP}(\mathbf{X}_p)$. As in the case of NBL, we assume that each attribute is independent of other attributes given the class.

Calculation of Class Conditional Frequency Counts

Let $A = \{A_1, A_2, \ldots, A_N\}$ be an ordered sequence of attribute names and $T = \{T_1, T_2, \ldots, T_n\}$ the corresponding set of AVTs. Let $C = \{c_1, c_2, \ldots, c_M\}$ is a set of mutually disjoint class labels. Let $\psi(v, T_i)$ be the set of descendents of a node corresponding to value v in a taxonomy T_i; *Children* (v, T_i), the set of all children – that is, direct descendents of a node corresponding to value v in a taxonomy T_i; $A(v, T_i)$ the list of ancestors, including the root, for v in T_i. Let $\sigma_i(v|c_j)$ be the frequency count of value v of attribute A_i given class label c_j in a training set D and $p_i(v|c_j)$, the estimated class conditional probability of value v of attribute A_i given class label c_j in a training set D.

Given an attribute value taxonomy T_i for attribute A_i, we can define a tree of class conditional frequency counts CCFC(A_i) such that there is an one-to-one correspondence between the nodes of the AVT T_i and the nodes of the corresponding CCFC(A_i). It follows that the class conditional frequency counts associated with a non leaf node of CCFC(A_i) should correspond the aggregation of the corresponding class conditional frequency counts associated with its children. Because each cut through an AVT T_i corresponds to a partition of the set of possible values $V(A_i)$ of the attribute A_i, the corresponding cut through CCFC(A_i) specifies a valid class conditional probability table for the attribute A_i. If all of the instances in the data set D are fully specified, estimation of CCFC(A_i) for each attribute is straightforward: We simply estimate the class conditional frequency counts associated with each of the primitive values of A_i from the data set D and use them recursively to compute the class conditional frequency counts associated with the non-leaf nodes of CCFC(A_i). When some of the data are partially specified, we can use a 2-step process for computing CCFC(A_i) [Zhang and Honavar, 2003]: First we make an upward pass aggregating the class conditional frequency counts based on the specified attribute values in the data set. Then we propagate the counts associated with partially specified attribute values down through the tree, augmenting the counts at lower levels according to the distribution of values along the branches based on the subset of the data for which the corresponding values are fully specified. This is a straightforward generalization of a standard approach to dealing with missing attribute values to the case of partially specified attribute values. The procedure is shown below.

1. Calculate frequency counts $\sigma_i(v|c_j)$ for each node v in T_i using the class conditional frequency counts associated with the specified values of attribute A_i in training set D.

2. For each attribute value v in T_i which received non-zero counts as a result of step (1), aggregate the counts upward from each such node v to its ancestors $A(v, T_i)$: $\sigma_i\left(w \mid c_j\right)_{w \in \Lambda(v,T_i)} \leftarrow \sigma_i\left(w \mid c_j\right) + \sigma_i\left(v \mid c_j\right)$

3. Starting from the root, recursively propagate the counts corresponding to partially specified instances at each node v downward according to the observed distribution among its children to obtain updated counts for each child $u_l \in$ Children (v, T_i):

$$\sigma_i\left(u_l \mid c_j\right) \leftarrow \sigma_i\left(u_l \mid c_j\right)\left(1 + \frac{\sigma_i\left(v \mid c_j\right) - \sum_{k=1}^{|Children(v,T_i)|} \sigma_i(u_k \mid c_j)}{\sum_{k=1}^{|Children(v,T_i)|} \sigma_i(u_k \mid c_j)}\right)$$

$$\text{if} \sum_{k=1}^{|Children(v,T_i)|} \sigma_i(u_k \mid c_j) \neq 0. \text{ Otherwise, } \sigma_i\left(u_l \mid c_j\right) \leftarrow \left(\frac{\sigma_i\left(v \mid c_j\right)}{|Children(v,T_i)|}\right)$$

Let $\Gamma = \{\gamma_1, \gamma_2, \ldots, \gamma_N\}$ be a set of cuts where ,γ_i stands for a cut through CCFC(A_i). The estimated conditional probability table CPT(γ_i) associated with the cut γ_i can be calculated from CCFC(A_i) using Laplacian estimator [Mitchell, 1997]

$$p_i\left(v \mid c_j\right)_{v \in \gamma_i} \leftarrow \frac{1 + \sigma_i\left(v \mid c_j\right)}{|\gamma_i| + \sum_{u \in \gamma_i} \sigma_i(u \mid c_j)}$$

The Naïve Bayes Classifier $h(\Gamma)$ based on a chosen set of cuts Γ is completely specified by the conditional probability tables associated with the cuts in Γ:

$$h(\Gamma) = \{\mathrm{CPT}(\gamma_1), \mathrm{CPT}(\gamma_2), \ldots, \mathrm{CPT}(\gamma_N)\}$$

If each cut $\gamma_i \in \Gamma$ is chosen to correspond to the primitive values of the respective attribute i.e., $\forall i$, $\gamma_i = Leaves(\text{CCFC}(A_i))$, $h(\Gamma)$ is simply the standard Naïve Bayes Classifier based on the attributes $A_1, A_2, ..., A_N.$

Searching for a Compact Naïve Bayes Classifier

We start with the Naïve Bayes Classifier that is based on the most abstract value of each attribute and successively refine the classifier using a criterion that is designed to tradeoff between the accuracy of classification and the complexity of the resulting Naïve Bayes classifier.

We say that a cut λ_i is a refinement of a cut γ_i if λ_i is obtained by replacing at least one attribute value $v \in \gamma_i$ by its descendents $A(v, T_i)$. We say that a set of cuts Δ is a refinement of Γ if at least one cut in Δ is a refinement of a cut in Γ. We say that a hypothesis $h(\Delta)$is a refinement of hypothesis $h(\Gamma)$ if Δ is a refinement of Γ. Figure 2 illustrates hypothesis refinement process.

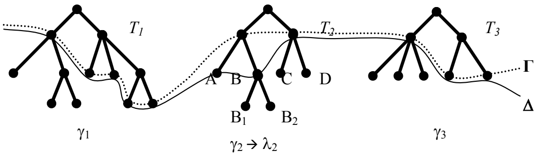

Figure 2. Hypothesis refinement. The cut $\gamma_2 = \{A, B, C, D\}$ in T_2 has been refined to $\lambda_2 = \{A, B_1, B_2, C, D\}$ by replacing B with its two children B_1, B_2. Therefore, $\Delta=\{\gamma_1, \lambda_2, \gamma_1\}$ is a refinement of $\Gamma=\{\gamma_1, \gamma_2, \gamma_1\}$, and corresponding hypothesis h(Δ) is a refinement of h(Γ).

The scoring function that we use to evaluate a candidate AVT-guided refinement of a Naïve Bayes Classifier is based on a variant of the minimum description length (MDL) score Rissanen [1978] which captures the tradeoff between the complexity and accuracy of the model. MDL score captures the intuition that the goal of a learner is to compress the training data D and encode it in the form of a hypothesis or a model h so as to minimize the length of the message that encodes the model h and the data D given the model h. Friedman et al (1997) suggested the use of a *conditional* MDL score in the case of hypotheses that are used for classification (as opposed to modeling the joint probability distribution of a set of random variables) to capture the tradeoff between the complexity and accuracy of the classifier:

$$CMDL(h \mid D) = \left(\frac{\log|D|}{2}\right) size(h) - CLL(h \mid D)$$

where

$$CLL(h \mid D) = |D| \sum_{p=1}^{|D|} \log P_h\left(c_p \mid a_{1p}, ..., a_{Np}\right)$$

where $P_h(c_p| \ a_{1p}, \ a_{2p}, \ .... \ a_{Np})$ denotes the conditional probability assigned to the class $c_p \in C$ associated with the training sample $\mathbf{X}_p = (a_{1p}, a_{2p}, ... a_{Np})$ by the classifier h, $size(h)$ is the number of parameters used by h, $|D|$ the size of the data set, and $CLL(h \ |D)$ is the conditional log likelihood of the data D given a hypothesis h. In the case of a Naïve Bayes classifier h, $size(h)$ corresponds to the total number of class conditional probabilities needed to describe h. Because each attribute is assumed to be independent of the others given the class in a Naïve Bayes classifier, we have:

$$CLL(h \mid D) = |D| \sum_{p=1}^{|D|} \log P_h\left(c_p \mid a_{1p}, ..., a_{Np}\right) = |D| \sum_{p=1}^{|D|} \log \left(\frac{P(c_p) \prod_i P_h(a_{ip} \mid c_p)}{\sum_{j=1}^{|C|} P(c_j) \prod_i P_h(a_{ip} \mid c_j)} \right)$$

where $P(c_p)$ is the prior probability of the class c_p which can be estimated from the observed class distribution in the data D.

There are two cases in the calculation of the conditional likelihood $CLL(h|D)$ when D contains partially specified instances. The first case is when a partially specified value of attribute A_i for an instance lies on the cut γ through $CCFC(A_i)$ or corresponds to one of the descendents of the nodes in the cut. In this case, we can treat that instance as though it were fully specified relative to the Naïve Bayes classifier based on the cut γ of $CCFC(A_i)$ and use the class conditional probabilities associated with the cut γ to calculate its contribution to $CLL(h|D)$. The second case is when a partially specified value (say v) of A_i is an ancestor of a subset (say λ) of the nodes in γ. In this case, we can aggregate the class conditional probabilities of the nodes in λ to calculate the contribution of the corresponding instance to $CLL(h|D)$.

Because each attribute is assumed to be independent of others given the class, the search for the AVT-based Naïve Bayes classifier (AVT-NBC) can be performed efficiently by optimizing the criterion independently for each attribute. This results in a hypothesis h that intuitively trades off the complexity of Naïve Bayes classifier (in terms of the number of parameters used to describe the relevant class conditional probabilities) against accuracy of classification. The algorithm terminates when none of the candidate refinements of the classifier yield statistically significant improvement in the CMDL score. The procedure is outlined below.

```
1. Initialize each $\gamma_i$ in $\Gamma=\{\gamma_1,\gamma_2,\ldots,\gamma_N\}$ to $\{Root(T_i)\}$
2. Estimate probabilities that specify the hypothesis $h(\Gamma)$.
3. For each cut $\gamma_i$ in $\Gamma=\{\gamma_1,\gamma_2,\ldots,\gamma_N\}$:
   A. Set $\delta_i \leftarrow \gamma_i$
   B. Until there are no updates to $\gamma_i$
      i. For each $v \in \delta_i$,
         a. Generate a refinement $\gamma^v_i$ of $\gamma_i$ by replacing $v$ with $Children(v, T_i)$, and refine $\Gamma$ accordingly to obtain $\Delta$. Construct corresponding hypothesis $h(\Delta)$.
         b. If $CMDL(h(\Delta)|D) < CMDL(h(\Gamma)|D)$, replace $\Gamma$ with $\Delta$ and $\gamma_i$ with $\gamma^v_i$
      ii. $\delta_i \leftarrow \gamma_i$
4. Output $h(\Gamma)$
```

5. Experiments and Results

5.1 Experiments

Our experiments were designed to explore the performance of AVT-NBL relative to that of the standard Naïve Bayes algorithm (NBL) and a Naïve Bayes Learner applied to a propositionalized version of the data set (PROP-NBL).

Although partially specified data and hierarchical AVT are common in many application domains, at present, there are few standard benchmark data sets of partially specified data and the associated AVT. Hence, we describe results of experiments with several data sets (MUSHROOM, SOYBEAN, AND NURSERY) adapted from the UC Irvine Repository. In the case of MUSHROOM TOXICOLOGY dataset, 17 of the 22 attributes have AVT supplied by a botanist. In the case of the SOYBEAN and NURSERY data sets, the AVTs for nominal attributes were specified based on our understanding of the domain.

The first set of experiments compares the performance of AVT-NBL, NBL, and PROP-NBL on the original (fully specified) data. The second set of experiments explores the performance of the three algorithms on data sets with different percentages of totally missing or partially missing attribute values. Data sets with a pre-specified percentage (0%, 10%, 30%, or 50%) of partially missing attribute values were generated by assuming that the missing values are uniformly distributed on the nominal attributes (see [Zhang and Honavar, 2003] for details). In each case, the error rate and the size (as measured by the number of class conditional probabilities used to specify the learned classifier) were estimated using 10-fold cross-validation.

5.2 Results

AVT-NBL yields significantly lower error rates than NBL and PROP-NBL on the original fully specified data.

Table 1 shows the estimated error rates of the classifiers generated by the AVT-NBL, NBL, and PROP-NBL on three benchmark data sets from UC Irvine Repository. The error rate of AVT-NBL is substantially smaller than that of NBL and PROP-NBL, with the difference in error rates being most pronounced in the case of MUSHROOM data. It is worth noting that PROP-NBL (NBL applied to a transformed data set using Boolean features that correspond to nodes of the AVTs) generally produces classifiers that have substantially higher error rates than AVT-NBL applied to the original data set. This can be explained by the fact that the Boolean features generated from an AVT are generally not independent given the class. This argues for the investigation of algorithms such as AVT-NBL based on principled ways of exploiting supplied by an AVT in generating classifiers.

Table 1. Comparison of error rates of **NBL**, **PROP-NBL** and **AVT-NBL** on benchmark data sets.

		NBL	PROP-NBL	AVT-NBL
MUSHROOM	0%	4.43%	4.45%	1.36%
	10%	4.65%	4.69%	1.46%
	30%	5.28 %	4.84%	1.57%
	50%	6.63%	5.82%	2.06%
NURSERY	0%	9.67%	10.59%	9.67%
	10%	15.27%	15.50%	12.97%
	30%	26.84%	26.25%	21.27%
	50%	36.96%	35.88%	29.34%
SOYBEAN	0%	7.03%	8.19%	6.44%
	10%	11.12%	11.13%	8.49%
	30%	12.45%	11.78%	8.99%
	50%	17.42%	14.91%	12.35%

Table 2. Comparison of the complexity of the classifiers as measured by the number of class conditional probabilities needed to specify the Naïve Bayes Classifier generated by **NBL**, **PROP-NBL** and **AVT-NBL** on benchmark data sets.

	NBL		PROP-NBL		AVT-NBL	
DATA SET	Percentage of Partially Missing Values					
	0%	50%	0%	50%	0%	50%
MUSHROOM	252	252	682	682	192	194
NURSERY	135	135	355	355	125	125
SOYBEAN	1900	1900	4959	4959	1653	1723

AVT-NBL yields classifiers that are substantially more compact than those generated by PROP-NBL and NBL.

Table 2 compares the total number of class conditional probabilities needed to specify the classifiers produced by AVT-NBL, NBL, and PROP-NBL when 0% and 50% of the attribute values are partially specified. The results show that AVT-NBL is effective in exploiting the information supplied by the AVT to generate accurate yet compact classifiers. Thus, AVT-guided learning algorithms offer an approach to compressing class conditional probability distributions that is different from the statistical independence-based factorization used in Bayesian Networks.

AVT-NBL yields significantly lower error rates than NBL and PROP-NBL on partially specified data.

Table 1 compares the estimated error rates of AVT-NBL with that of NBL and PROP-NBL in the presence of varying percentages of (10%, 30% and 50%) of partially missing attribute values. Naïve Bayes classifiers generated by AVT-NBL have substantially lower error rates than those generated by NBL and PROP-NBL, with the differences being more pronounced at higher percentages of partially missing attribute values.

6. Summary and Discussion

6.1 Summary

In this paper, we have presented AVT-NBL, an algorithm for learning Naïve Bayes Classifiers using attribute value taxonomies from partially specified data. AVT-NBL is a natural generalization of the standard algorithm (NBL) for learning Naïve Bayes Classifiers.

Experimental results presented in the paper show that:

(1) AVT-NBL is able to learn substantially more accurate Naïve Bayes classifiers than those produced by NBL and PROP-NBL from data sets with varying percentages of partially specified attribute values (including data sets with no partially specified attribute values).

(2) Classifiers generated by AVT-NBL are substantially more compact than those generated by NBL and PROP-NBL.

6.2 Related Work

There is some work in the machine learning community on the problem of learning classifiers from attribute value taxonomies (sometimes called tree-structured attributes) and *fully* specified data in the case of decision trees and rules (see Zhang and Honavar, 2003 for a review) desJardins et al [2000] suggested the use of Abstraction-Based Search (ABS) to learn Bayesian networks with compact structure. Zhang and Honavar [2003] describe AVT-DTL, an efficient algorithm for learning decision tree classifiers from AVT and partially specified data. With the exception of AVT-DTL, to the best of our knowledge, there are no algorithms for learning classifiers from AVT and partially specified data.

There has been some work on the use of class taxonomy (CT) in the learning of classifiers in scenarios where class labels correspond to nodes in a predefined class hierarchy [Clare and King, 2001; Koller and Sahami, 1997].

MDL principle has been used to learn unrestricted Bayesian belief networks [Lam and Bacchus, 1994; Suzuki, 1998]. Friedman et al. [1997] suggested the use of class conditional MDL (CMDL) score for constructing Bayesian classifiers. In general, computation of CMDL score is not computationally feasible. The AVT-NBL algorithm described in this paper demonstrates the use of CMDL score to guide AVT-based search for compact and accurate Naïve Bayes classifiers.

There is a large body of work on the use of domain theories to guide learning. AVT can be viewed as a restricted class of domain theories. However, the work on exploiting domain theories in learning has not focused on the effective use of AVT to learn classifiers from partially specified data.

Chen and Tseng [1996] proposed database models to handle imprecision using partial values and associated probabilities where a partial value refers to a set of possible values for an attribute. McClean et al [2001] proposed aggregation operators defined over partial values. While this work suggests ways to aggregate statistics so as to minimize information loss, it does not address the problem of learning from AVT and partially specified data.

Automated construction of hierarchical taxonomies over attribute values and class labels is beginning to receive attention in the machine learning community. Examples include distributional clustering, [Pereira et al., 1993], extended FOCL and statistical clustering [Yamazaki et al., 1995], information bottleneck [Slonim & Tishby 2000]. Such algorithms provide a source of AVT in domains where none are available. However, the focus of work described in this paper is on algorithms that use AVT in learning classifiers from data.

6.3 Future Work

Some directions for future work include:

(1) Development AVT-based variants of other machine learning algorithms for construction of classifiers from partially specified data from distributed, semantically heterogeneous data sources [Reinoso-Castillo et al., 2003; Caragea et al., 2004].

(2) Extension of the algorithms like AVT-DTL and AVT-NBL to handle taxonomies defined over ordered and numeric attribute values.

(3) Further experimental evaluation of AVT-NBL, AVT-DTL, and related learning algorithms on a broad range of data sets in scientific knowledge discovery applications e.g., computational biology.

References

Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, Davis AP, Dolinski K, Dwight SS, Eppig JT, Harris MA, Hill DP, Issel-Tarver L, Kasarskis A, Lewis S, Matese JC, Richardson JE, Ringwald M, Rubin GM, Sherlock G. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. Nat Genet. 2000 25(1):25-9.

Berners-Lee, T., Hendler, J. and Ora Lassila. The semantic web. *Scientific American*, May 2001.

Caragea, D., Silvescu, A., and Honavar, V. (2004). A Framework for Learning from Distributed Data Using Sufficient Statistics and its Application to Learning Decision Trees. *International Journal of Hybrid Intelligent Systems.* Vol. 1. pp. 80-89.

Chen, A.L.P., Tseng, F.S.C. (1996) Evaluating aggregate operations over imprecise data. IEEE Trans. On Knowledge and Data Engineering, 8, 273-284.

Clare, A. and King, R.D. (2001). Knowledge Discovery in Multi-label Phenotype Data. In: Lecture Notes in Computer Science. Vol. 2168, pp. 42-, 2001.

desJardins, M., Getoor, L. & Koller, D. (2000). Using Feature Hierarchies in Bayesian Network Learning. In: Proceedings of the Symposium on Abstraction, Reformulation, Approximation. Lecture Notes in Artificial Intelligence 1864: 260-270. Springer-Verlag

Friedman, N., Geiger, D., Goldszmidt, M. (1997) Bayesian Network Classifiers. Machine Learning, Vol: 29, 1997, 131-163

Haussler, D. (1988). Quantifying Inductive Bias: AI Learning Algorithms and Valiant's Learning Framework. *Artificial Intelligence*, 36:177-221

Kohavi, R. and Provost, P. (2001). Applications of Data Mining to Electronic Commerce. Data Mining and Knowledge Discovery, pp. Vol. 5., pp. 1-7. 2001.

Koller, D. and Sahami, M. (1997). Hierarchically classifying documents using very few words. In: Proceedings of the 1997 Conference on Machine Learning, San Mateo, CA: Morgan Kaufmann. pp. 170-178.

Lam, W., Bacchus, F. (1994). Learning Bayesian Belief Networks An approach based on the MDL Principle. Computational Intelligence, Vol 10:4, 1994

Langley, P., Iba, W., & Thompson, K. (1992). An analysis of Bayesian classifiers Proceedings of the Tenth National Conference on Artificial Intelligence (pp. 223-228). San Jose, CA: AAAI Press.

Martin, F.J. and Plaza, E. *SOID: A Simple Ontology for Intrusion Detection*. Seventh International Conference on Knowledge-Based Intelligent Information & Engineering Systems, KES'2003, Oxford, United Kingdom, number 2773 in Lecture Notes in Artificial Intelligence. Springer-Verlag, pp 1222-1229, September 2003.

McCallum, A., Rosenfeld, R., Mitchell, T., Ng. A. (1998) Improving Text Classification by Shrinkage in a Hierarchy of Classes. Proceedings of the Fifteenth International Conference on Machine Learning.

McClean, S., Bryan W. Scotney, Mary Shapcott Aggregation of Imprecise and Uncertain Information in Databases. *IEEE Transactions on Knowledge and Data Engineering* (6): 902-912 (2001).

Mitchell, T. (1997). Machine Learning. New York: Addison-Wesley.

Pazzani, M., Mani, S., & Shankle, W.R. (1997). *Beyond concise and colorful: Learning Intelligible Rules*. In Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining, 235-238. Newport Beach, CA: AAAI Press.

Pereira, F., Tishby, N. & Lee, L. (1993). Distributional clustering of English words. In: Proceedings of the Thirty-first Annual Meeting of the Association for Computational Linguistics, pp. 183-190.

Reinoso-Castillo, J., Silvescu, A., Caragea, D., Pathak, J., and Honavar, V. (2003) A Federated Query Centric Approach to Information Extraction and Integration from Heterogeneous, Distributed, and Autonomous Data Sources. In: The 2003 IEEE International Conference on Information Reuse and Integration, October 27-29, 2003, Las Vegas, USA. IEEE Press.

Rissanen, J. (1978). Modeling by shortest data description. *Automatica,* vol. 14, 1978, pp. 465-471.

Slonim, N., Tishby, N. (2000). Document Clustering using Word Clusters via the Information Bottleneck Method. ACM SIGIR, pp 208-215, 2000.

Sowa, J. (1999) Knowledge Representation: Logical, Philosophical, and Computational Foundations. New York: PWS Publishing Co.

Suzuki, J. (1996) Learning Bayesian Belief Networks Based on the MDL Principle: An Efficient Algorithm Using the Branch and Bound Technique. Proceedings of the Thirteenth International Conference on Machine Learning.

Undercoffer, J., Anupam Joshi, Tim Finin, and John Pinkston. A Target Centric Ontology for Intrusion Detection: Using DAML+OIL to Classify Intrusive Behaviors. To appear, Knowledge Engineering Review - Special Issue on Ontologies for Distributed Systems, Cambridge University Press, 2004.

Yamazaki, T., Pazzani, M. & Merz, C. (1995). Learning Hierarchies from Ambiguous Natural Language Data. In: Proceedings of the Twelfth International Conference on Machine Learning. pp. 329-342. Morgan-Kaufmann.

Zhang, J., Silvescu, A., and Honavar, V. (2002). Ontology-Driven Induction of Decision Trees at Multiple Levels of Abstraction. *Proceedings of Symposium on Abstraction, Reformulation, and Approximation 2002.* Vol. 2371 of Lecture Notes in Artificial Intelligence : Springer-Verlag.

Zhang, J. and Honavar, V. (2003). Learning From Attribute Value Taxonomies and Partially Specified Instances. In: Proceedings of the International Conference on Machine Learning (ICML 2003). pp. 880-887. AAAI Press.