

# A Bayesian Framework for Hierarchical Classification

Nicolò Cesa-Bianchi      Alex Conconi  
DTI, Università degli Studi di Milano, Italy  
{cesa-bianchi, conconi}@dti.unimi.it

Claudio Gentile  
Dipartimento di Informatica e Comunicazione  
Università dell'Insubria, Italy  
gentile@dsi.unimi.it

## 1 Introduction

We investigate the problem of classifying data based on the knowledge that the graph of dependencies between class elements is a tree forest. The trees in the forest are collectively interpreted as a taxonomy. That is, we assume that every data instance is labelled with a (possibly empty) set of class labels and, whenever an instance is labelled with a certain label  $\mathbf{i}$ , then it is also labelled with all the labels on the path from the root of the tree where  $\mathbf{i}$  occurs down to node  $\mathbf{i}$ . We also allow for multiple-path labellings, where instances can be tagged with labels belonging to two or more paths in the forest.

Given a taxonomy, we learn a hierarchical classifier by fitting the training data with the parameters of a Bayesian network defined on the taxonomy. We show a practical algorithm for learning Bayesian networks of this form. In the full paper we report the results of hierarchical classification experiments carried out on the Reuters corpus of newswire stories.

Finally, we show how our approach can be easily extended to a very general Bayesian framework for learning multilabelled data.

## 2 A Bayesian model

We assume that instances are encoded as attribute vectors  $\mathbf{x} = (x_1, \dots, x_d) \in \mathbb{R}^d$ . A *multilabel* for an instance  $\mathbf{x}$  is any subset of the set  $\{1, \dots, \ell\}$  of all labels, including the empty set. We represent the multilabel of  $\mathbf{x}$  with a vector  $\mathbf{v} = (v_1, \dots, v_\ell) \in \{-1, 1\}^\ell$ , where  $\mathbf{i}$  belongs to the multilabel of  $\mathbf{x}$  iff  $v_i = 1$ . A *taxonomy* is a forest whose trees are defined over the set of labels. A multilabel  $\mathbf{v}$  belongs to a given taxonomy iff it is the union of one or more paths in the forest, where each path must start from a root but need not terminate on a leaf (see Figure 1).

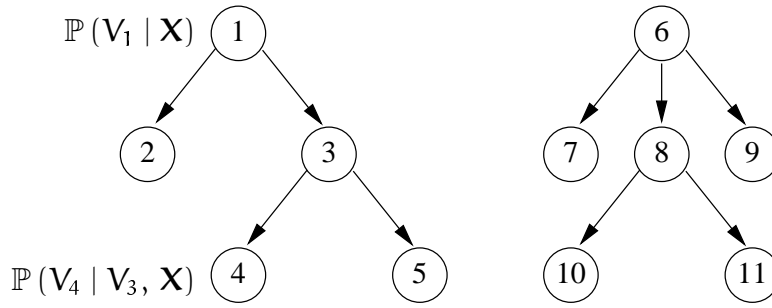


Figure 1: A forest made up of two disjoint trees. The nodes are tagged with the name of the labels, so that in this case  $\ell = 11$ . According to our definition, the multilabel  $\mathbf{v} = (1, -1, 1, -1, -1, 1, -1, 1, -1, 1, -1)$  belongs to this taxonomy (since it is the union of paths  $1 \rightarrow 3$  and  $6 \rightarrow 8 \rightarrow 10$ ), while the multilabel  $\mathbf{v} = (1, 1, -1, 1, -1, -1, -1, -1, -1, -1, -1)$  does not, since  $1 \rightarrow 2 \rightarrow 4$  is not a path in the forest.

In order to illustrate the probabilistic model generating the examples, we temporarily switch to a more general (and well-known) Bayesian network model.

Assume we have a Bayesian network  $\mathbf{G}$  specified by a directed acyclic graph (DAG) whose nodes are the labels  $1, \dots, \ell$ . Let  $\text{PAR}(\mathbf{i})$  be the set of parent nodes of  $\mathbf{i}$ . Each node  $\mathbf{i}$  of  $\mathbf{G}$  is tagged with a  $\{-1, 1\}$ -valued random variable  $\mathbf{V}_i$  and with a conditional probability function of the form  $\mathbb{P}(\mathbf{V}_i = \mathbf{v}_i \mid \{\mathbf{V}_j = \mathbf{v}_j : j \in \text{PAR}(\mathbf{i})\}, \mathbf{X} = \mathbf{x})$ . Hence, the quantity

$$f_{\mathbf{G}}(\mathbf{v} \mid \mathbf{x}) = \prod_{i=1}^{\ell} \mathbb{P}(\mathbf{V}_i = \mathbf{v}_i \mid \{\mathbf{V}_j = \mathbf{v}_j : j \in \text{PAR}(\mathbf{i})\}, \mathbf{X} = \mathbf{x})$$

defines a joint probability distribution on  $\mathbf{V}_1, \dots, \mathbf{V}_{\ell}$  conditioned on  $\mathbf{x}$  being the current instance. We call the pair  $(\mathbf{x}, \mathbf{v})$  an example, where  $\mathbf{v}$  is the multilabel of  $\mathbf{x}$ .

Recall that a Bayesian network  $\mathbf{G}$  defines a *maximum likelihood classifier*  $\mathbf{h}_{\mathbf{G}}$  given by

$$\mathbf{h}_{\mathbf{G}}(\mathbf{x}) = \operatorname{argmax}_{\mathbf{v} \in \{-1, 1\}^{\ell}} f_{\mathbf{G}}(\mathbf{v} \mid \mathbf{x}) . \quad (1)$$

Note that in the above formula  $\mathbf{x}$  is given, hence  $\mathbf{h}_{\mathbf{G}}(\mathbf{x})$  is a maximum likelihood classifier just over the labels (given  $\mathbf{x}$ ). Note further that for each instance  $\mathbf{x}$ ,  $\mathbf{h}_{\mathbf{G}}(\mathbf{x})$  can be computed using, e.g., the sum-product algorithm for graphical models [4].

In the hierarchical learning model studied in this paper, the multilabel  $\mathbf{v}$  of each instance  $\mathbf{x}$  is generated via a random draw from the joint distribution  $f_{\mathbf{G}}(\cdot \mid \mathbf{x})$  defined by a fixed Bayesian network  $\mathbf{G}$  restricted to a taxonomy. In fact, given an arbitrary taxonomy, we can easily define on it a Bayesian network  $\mathbf{G}$  whose joint distribution only generates multilabels that belong to the taxonomy. The only constraint we have to enforce on  $\mathbf{G}$  is that

$$\mathbb{P}(\mathbf{V}_i = 1 \mid \mathbf{V}_j = -1, \mathbf{X} = \mathbf{x}) = 0$$

where  $j$  is the unique parent of  $\mathbf{i}$ . We call this  $\mathbf{G}$  a *taxonomy-based* Bayesian network. For instance, in the taxonomy-based Bayesian network of Figure 1 we have  $\mathbb{P}(\mathbf{V}_4 \mid \mathbf{V}_3 = -1, \mathbf{X}) = 0$  for all values of  $\mathbf{X}$  and  $\mathbf{V}_4$ .

Our stochastic model is thus defined by a pair  $(\mathbf{D}, \mathbf{G})$ , where  $\mathbf{D}$  is a distribution on  $\mathbb{R}^d$  and  $\mathbf{G}$  is a taxonomy-based Bayesian network  $\mathbf{G}$ . This model specifies the data-generating i.i.d. process  $\{(\mathbf{X}_1, \mathbf{V}_1), (\mathbf{X}_2, \mathbf{V}_2), \dots\}$ , where each  $\mathbf{X}_t$  is distributed according to a fixed and unknown distribution  $\mathbf{D}$  and each  $\mathbf{V}_t$  is distributed according to  $f_{\mathbf{G}}(\cdot | \mathbf{X}_t)$ .

### 3 Learning hierarchical classifiers

We now introduce a parametric model for taxonomy-based Bayesian networks. To each node  $i$  in the taxonomy, we associate a unit-norm weight vector  $\mathbf{u}_i \in \mathbb{R}^d$ . Then, we define the conditional probabilities for a non-root node  $i$  with parent  $j$  as follows:

$$\mathbb{P}(V_i = 1 | V_j = 1, \mathbf{X} = \mathbf{x}) = \frac{1 + \mathbf{u}_i^\top \hat{\mathbf{x}}}{2},$$

where  $\hat{\mathbf{x}} = \mathbf{x}/\|\mathbf{x}\|$ . If  $i$  is a root node, the above simplifies to

$$\mathbb{P}(V_i = 1 | \mathbf{X} = \mathbf{x}) = \frac{1 + \mathbf{u}_i^\top \hat{\mathbf{x}}}{2}.$$

Given the knowledge of the underlying forest, we learn an estimate  $\hat{\mathbf{G}}$  of a taxonomy  $\mathbf{G}$ . Then, we use the hypothesis  $h_{\hat{\mathbf{G}}}$  in (1) to hierarchically classify new data. To learn  $\hat{\mathbf{G}}$  we independently learn each  $\mathbf{u}_i$  via the second-order Perceptron algorithm [1, 2], an incremental algorithm for learning probabilistic linear-threshold classifiers.

### 4 Experiments

We plan to test our hierarchical classification algorithm on the Reuters corpus volume 1 (RCV1) consisting in over 800K documents classified in a taxonomy of 101 nodes.

### 5 Extensions

By considering general Bayesian networks (i.e., not necessarily taxonomy-based), one obtains a model for generating multilabelled data allowing arbitrary stochastic dependencies among labels. We are currently investigating the problem of learning such networks by extending to multilabelled data the framework of Bayesian network classifiers [3].

### References

- [1] N. Cesa-Bianchi, A. Conconi, and C. Gentile. A second-order Perceptron algorithm. In *Proceedings of the 15th Annual Conference on Computational Learning Theory*, pages 121–137. LNAI 2375, Springer, 2002.
- [2] N. Cesa-Bianchi, A. Conconi, and C. Gentile. Learning probabilistic linear-threshold classifiers via selective sampling. In *Proceedings of the 16th Annual Conference on Learning Theory*, pages 373–386. LNAI 2777, Springer, 2003.

- [3] N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian network classifiers. *Machine Learning*, 29(2-3):131–163, 1997.
- [4] F.R. Kschischang, B.J. Frey, and H.-A. Loeliger. Factor graphs and the sum-product algorithm. *IEEE Transactions on Information Theory*, 47(2):498–519, 2001.