Noname manuscript No.
(will be inserted by the editor)

Optimization of Robust Loss Functions for Weakly-Labeled Image Taxonomies

Julian J. McAuley · Arnau Ramisa · Tibério S. Caetano

the date of receipt and acceptance should be inserted later

Abstract The recently proposed *ImageNet* dataset consists of several million images, each annotated with a single object category. These annotations may be imperfect, in the sense that many images contain *multiple* objects belonging to the label vocabulary. In other words, we have a multi-label problem but the annotations include only a single label (which is not necessarily the most prominent). Such a setting motivates the use of a *robust* evaluation measure, which allows for a limited number of labels to be predicted and, so long as one of the predicted labels is correct, the overall prediction should be considered correct. This is indeed the type of evaluation measure used to assess algorithm performance in a recent competition on ImageNet data. Optimizing such types of performance measures presents several hurdles even with existing structured output learning methods. Indeed, many of the current state-of-the-art methods optimize the prediction of only a single output label, ignoring this 'structure' altogether. In this paper, we show how to directly optimize continuous surrogates of such performance measures using structured output learning techniques with latent variables. We use the output of existing binary classifiers as input features in a new learning stage which optimizes the structured loss corresponding to the robust performance measure. We present empirical evidence that this allows us to 'boost' the performance of binary classification on a variety of weakly-supervised labeling problems defined on image taxonomies.

Keywords Image Labeling · Image Tagging · Image Taxonomies · Structured Learning

1 Introduction

The recently proposed *ImageNet* project consists of building a growing dataset of images, organized into a taxonomy based on the WordNet hierarchy (Deng et al., 2009). Each node in this taxonomy includes a large set of images (in the hundreds or thousands). From an object recognition point of view, this dataset is interesting because it naturally suggests the possibility of leveraging the image taxonomy in order to improve recognition beyond what can be achieved independently for each image. Indeed this question has been the subject of much interest recently, culminating in a competition in this context using ImageNet data (Berg et al., 2010; Lin et al., 2011; Sánchez and Perronnin, 2011).

Each image in ImageNet may contain several objects from the label vocabulary, however the annotation includes only a single label per image, and this label is not necessarily the most prominent. This 'imperfect' annotation suggests that a meaningful performance measure in this dataset should somehow not penalize predictions that contain legitimate objects that are missing from the annotation. One way to deal with this issue is to use a *robust* performance measure based on the following idea: an algorithm is allowed to predict more than one label per image (up to a maximum of K labels, so that the solution is not degenerate), and so long as at least one of those labels agrees with the ground-truth label, no penalty is incurred. This is precisely the type of performance measure used to evaluate algorithm performance in the aforementioned competition (Berg et al., 2010).

Another form of 'weak' labeling that one typically observes in image datasets is the set of *tags* associated with an image, i.e., annotations provided by the community of

J. McAuley
InfoLab, Stanford University

A. Ramisa
Institut de Robòtica i Informàtica Industrial (CSIC-UPC)

T. Caetano
Machine Learning Group, NICTA, and the Australian National University

users on image hosting websites such as *Flickr*. As with ImageNet data, one observes only positive labels, i.e., we only observe whether an image *wasn't* assigned a particular tag, not whether it *couldn't* have been. This suggests that similar performance measures could be used to train a system for tag recommendation: it is sufficient that *one of* the suggested tags is similar to *one of* the groundtruth tags, though to our knowledge, this type of robust, hierarchical performance measure has not been applied to tag prediction.

In this paper, we present an approach for directly optimizing a continuous surrogate of these robust performance measures. In other words, we try to optimize the very measure that is used to assess recognition quality in the ImageNet 2010 Challenge dataset. We show empirically that by using binary classifiers as a starting point, which are state-of-the-art for this task, we can boost their performance by means of optimizing the structured loss. We also apply a variant of the same performance measure to the problem of tag recommendation, using a recently proposed dataset derived from Flickr images (Huiskes et al., 2010).

Essentially, we use latent variables to 'strengthen' the weakly labeled groundtruth. Intuitively, our latent variables are designed to represent those objects that appear in an image, but were not annotated. Given that the problem becomes one of fully-supervised structured learning when the latent variables are observed, we can use recently proposed techniques on structured learning with latent variables (Yu and Joachims, 2009) to simultaneously optimize the latent variables and the model parameters.

In addition to experiments on the ImageNet 2010 Challenge dataset, we study labeling problems on two other image taxonomies: the *MIR Flickr Retrieval Evaluation* (MIR, Huiskes and Lew, 2008), and the *ImageCLEF Annotation Task* (ImageCLEF, Nowak et al., 2011). From the former dataset we also obtain tag information from Flickr. The label vocabularies in MIR and ImageCLEF are much smaller than that of the ImageNet 2010 Challenge dataset (24 and 99 labels, respectively), meaning that taxonomic information is not typically used in training or evaluation on these datasets. However, they are useful in the sense that they allow us to study the behaviour of the latent variables mentioned above: since these datasets are fully annotated, we can 'pretend' that they are weakly labeled by withholding part of the annotation, allowing us to compare the predicted values of the latent variables with those of the withheld annotation.

Our experiments reveal that our latent variable model is beneficial for learning in all four of the taxonomies we examine.

An initial version of this paper appeared in McAuley et al. (2011).

1.1 Literature Review

The success of visual object classification achieved in recent years is pushing computer vision research towards more difficult goals in terms of the number of object classes and the size of the training sets used. For example, Perronnin et al. (2010) used increasingly large training sets of Flickr images together with online learning algorithms to improve the performance of linear SVM classifiers trained to recognize the 20 Pascal Visual Object Challenge 2007 objects; or Torralba et al. (2008), who defined a gigantic dataset of 75,062 classes (using all the nouns in WordNet) populated with 80 million tiny images of only 32 × 32 pixels. The WordNet nouns were used in seven search engines, but without any manual or automatic validation of the downloaded images. Despite their low resolution, the images were shown to still be useful for classification.

Similarly, Deng et al. (2009) created ImageNet: a vast dataset with thousands of classes and millions of images, also constructed by taking nouns from the WordNet taxonomy. These were translated into different languages, and used as query terms in multiple image search engines to collect a large amount of pictures. However, as opposed to the case of the previously mentioned 80 Million Tiny Images dataset, in this case the images were kept at full resolution and the labels were manually verified using Amazon Mechanical Turk. Currently, the full ImageNet dataset consists of over 17,000 classes and 12 million images. Figure 1 shows a few example images from various classes.

Deng et al. (2010) performed classification experiments using a substantial subset of ImageNet, including more than ten thousand classes and nine million images. Their experiments highlighted the importance of algorithm design when dealing with such quantities of data, and showed that methods believed to be better in small scale experiments turned out to under-perform when brought to larger scales. Also a cost function for classification taking into account the hierarchy was proposed. In contrast with Deng et al. (2010), most of the works using ImageNet for large scale classification make no use of its hierarchical structure.

As mentioned before, in order to encourage large scale image classification using ImageNet, a competition using a subset of 1,000 classes and 1.2 million images, called the ImageNet Large Scale Visual Recognition Challenge (Berg et al., 2010), was conducted together with the 2010 Pascal Visual Object Challenge competition. Notoriously, the better classified participants of the competition used traditional one-versus-all approaches and completely disregarded the taxonomic information.

Lin et al. (2011) obtained the best score in the competition using a conventional one-vs-all approach. Two state-of-the-art coding and pooling techniques, Local Coordinate Coding and Super-Vector Coding, were used to construct the

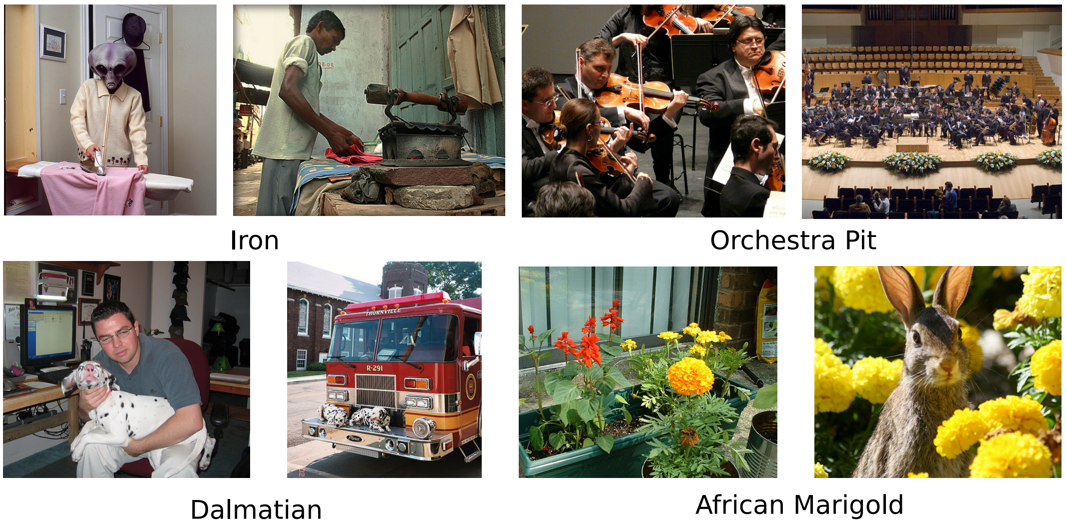


Fig. 1 Example images from ImageNet. Classes range from very general to very specific, and since there is only one label per image, it is not rare to find images with unannotated instances of other classes from the dataset.

descriptor vectors for each image. Finally, averaged stochastic gradient descent (ASGD) was used to efficiently train a thousand linear SVM classifiers.

Sánchez and Perronnin (2011) got the second best score in the competition. In their approach, they used high-dimensional Fisher Kernels for image representation with lossy compression techniques: first, dimensionality reduction using Hash Kernels (Shi et al., 2009) was attempted and secondly, since the results degraded rapidly with smaller descriptor dimensionality, coding with Product Quantizers (Jégou et al., 2010) was used to retain the advantages of a high-dimensional representation without paying a high price in terms of memory and I/O usage. To train the standard binary one-vs-all linear classifiers, they also used Stochastic Gradient Descent.

The difficulty of using hierarchical information for improving classification may be explained by the findings of Russakovsky and Fei-Fei (2010). They showed that in ImageNet, the relationships endowed by the WordNet taxonomy do not necessarily correspond to visual similarity, and that in fact new relations based only on visual appearance information can be established between some classes, possibly far away in the hierarchy.

In contrast with the findings of Russakovsky and Fei-Fei (2010), Deselaers and Ferrari (2011) experimentally validated, to a large degree, the common assumptions that *semantic categories are visually separable* and that *visual similarity is correlated with semantic similarity* in the ImageNet dataset; this was achieved by comparing the visual variability of images within a particular class, measured using GIST signatures. They also studied the relationship between semantic and visual distance, and proposed an image distance measure based on ImageNet data, termed the *ImageNet Distance* in their paper, to assess whether two images contain an instance of the same base-level category. This distance measures the visual and semantic similarity between the *categories* associated with the nearest neighbors in ImageNet of the images being compared.

A different image distance, also based on the ImageNet hierarchy, was proposed by Deng et al. (2011). There, the authors exploit semantic knowledge in the form of a hierarchy to compute a similarity measure for large scale same-category image retrieval. Provided that training data is available on the nodes of the hierarchy, classifiers are learned and mapped to probability values. Then, the class probabilities for two images can be compared using a 'cost' matrix that penalizes pairs of classes that have their first common ancestor higher in the hierarchy.

Kim et al. (2011) proposed a method to decompose an image descriptor into a sparse mixture of training 'base' descriptors to incorporate hierarchical information. The class labels of the training descriptors active in the mixing weights for the query descriptor can be seen as labels for the image.

Cai and Hofmann (2004) defined a taxonomy over categories in a text classification task, and showed that optimizing a structured loss defined on this taxonomy can improve performance. Binder et al. (2011) discussed similar learning methods for multiclass image classification, where training classes and examples are organized in a pre-defined taxonomy. Local SVM learning methods (where descendants of a node are used as positive examples, and other nodes are treated as negative examples) were shown to obtain similar performance to structured SVMs, while requiring less training time and being highly parallelizable. While not always improving upon the flat classifiers in terms of the 0/1 loss, the taxonomy-based classifiers consistently achieved lower hierarchical error, which translates into more meaningful, 'human-like', confusions. Multilabel experiments were also performed in Binder et al. (2011), but always with fully annotated images and in datasets with about 20 classes.

Blaschko et al. (2010) successfully exploited weakly annotated data to improve the performance of an object detection method framed in a structured output formulation. Annotations indicated the presence or absence of objects in images and weak information about object location. The locations of bounding boxes were treated as latent variables to be inferred during training, constrained by the type of annotation for each image. With this approach and a single full annotation, they were able to attain performance comparable to that obtained with complete annotation in the INRIA pedestrian dataset.

Other works aim to learn hierarchies directly from image information. Marszałek and Schmid (2008) proposed a method to construct a Relaxed Hierarchy DAG which avoids introducing aliasing due to hard-assigning a class to a particular branch of the tree, instead postponing the decision until fewer classes are present by assigning it to both branches. With this approach they show performance comparable to one-versus-all classification, while being sub-linear in complexity. Bart et al. (2008) proposed a generative modeling approach similar to Latent Dirichlet Allocation (LDA) for taxonomy learning. A category is defined as a mixture of topics, each representing certain features (like sea, sand, fur, sky, etc.). Then, categories are arranged in a tree structure by performing inference efficiently with a nonparametric prior over tree structures similar to a nested Chinese restaurant process, as used in text modeling. Qualitative results are given for the Corel dataset, and the method is shown to improve with respect to LDA in the 13-scenes dataset. Simultaneously (and independently), Sivic et al. (2008) also used a hierarchical LDA model with a nested Chinese restaurant process prior to discover a taxonomy of categories from visual information without supervision, and also showed improvements with respect to plain LDA in a pixel-level segmentation task using the MSRC-B1 dataset.

As mentioned in the introduction, the problem of tag prediction has many parallels with multi-label classification. In the following paragraphs we review some recent work on tag prediction.

Verbeek et al. (2010) used the MIR Flickr dataset to evaluate their previously proposed TagProp algorithm (Guillaumin et al., 2009). TagProp is a weighted nearest-neighbor model that propagates tag terms among images in a dataset to obtain a more complete annotation. Different visual features (e.g. SIFT, color histograms) as well as textual features derived from the tags were used. When compared to standard SVMs, the TagProp model performed worse when precise manual annotations were used, but better when using noisy Flickr tags as training labels. Tag-derived features proved beneficial in terms of accuracy both for SVMs and for TagProp.

Dimitrovski et al. (2010) performed hierarchical classification on the ImageCLEF dataset using a random forest approach. An ensemble of Predictive Clustering Trees were learned using multiple types of features. The tags of a novel query image are determined by propagating it down the trees and averaging the tag probabilities of the selected leaf from each tree of the forest. Despite the simplicity of the approach, it achieved the second position in the 2010 ImageCLEF photo annotation competition out of twelve competing groups.

Bucak et al. (2011) addressed the problem of incomplete annotations in the context of multi-label learning. They proposed a group-lasso based method to train a multi-label model that predicts a ranking of classes given a test image. The method was shown to attain results better than those of a standard SVM in the 2007 Pascal Visual Object Challenge, the MIR, and the ESP Game datasets.

Mensink et al. (2011) proposed a label prediction system that uses structured models to learn the dependencies among image labels. In an interactive setting, a small amount of user input can be used to significantly improve classification results by selectively asking questions to the user that minimize uncertainty in the remaining labels. The proposed models also improve the results of plain SVM classifiers in a non-interactive (i.e., automatic) setting, although the performance gain is modest. The system was tested in ImageCLEF, as well as in the SUN09 and in the Animals with Attributes datasets.

Wang et al. (2011) proposed a semi-supervised image annotation method that uses a bi-relational graph. The graph can be divided into a label correlation subgraph and an image similarity subgraph, with an additional bipartite subgraph defined by class assignments to images. A random walk with restarts was used to learn class-to-class and class-to-image relevances. In contrast to related work, asymmetric relationships between classes are considered (e.g. the probability of *road* given *car* is not the same as *car* given *road*). Finally, a method to learn these bi-directional probabilities was proposed and shown to perform better than the symmetric version.

Moran and Lavrenko (2011) modified the Continuous-Space Relevance Model of Lavrenko et al. (2003). Rather than using the top-ranked tags for an image, a set of tags is predicted jointly: their approach increases the probability of predicting less likely (but consistent) tags and reduces that of predicting irrelevant or contradictory, but highly scored, ones.

Other works exploited tags as a form of weak supervision, using them to complement purely visual information for image classification.

Guillaumin et al. (2010) showed how images with associated tags, but for which reliable labels are not known, can be used to complement a potentially smaller set of images with both tags and labels and, ultimately, compute better visual classifiers. The motivation of their work comes from the understanding that classifiers that exploit image and (weak) textual information significantly outperform those based on visual features alone, which makes them suitable for use in semi-supervised learning scenarios. This technique facilitates training on the large amounts of images available from online photo sharing sites for which expensive label information is not available.

Kawanabe et al. (2011) proposed kernels tailored to tags associated with Flickr images to complement and improve visual-feature-based image classification. The authors build on the 'tag kernel' proposed by Guillaumin et al. (2010) and address the issue of sparsity in tag-based feature representations by smoothing using Markov Random Walks over the tags. They showed a small but statistically significant improvement (according to the Wilcoxon test) over the original tag kernel formulation.

2 Problem Statement

Our notation is summarized in Table 1. We are given the dataset $\mathcal{S} = \left\{(x^1, Y^1), \ldots, (x^N, Y^N)\right\}$, where $x^n \in \mathcal{X}$ denotes an F-dimensional feature vector representing an image with a set of groundtruth labels Y^n.[1] Our goal is to learn a classifier $\bar{Y}(x;\theta)$ that for an image x outputs a set of K distinct object categories. The vector θ parameterizes the classifier $\bar{Y}$; we wish to learn θ so that the labels predicted by $\bar{Y}(x^n;\theta)$ are 'similar to' the training labels Y^n under some loss function $\Delta(\bar{Y}(x^n;\theta), Y^n)$. Our specific choice of classifier and loss function shall be given in Section 2.1. In short, the goal and contribution of this paper is to learn the classifier $\bar{Y}$ for precisely the loss function Δ that is used to measure performance in the ImageNet Large Scale Visual Recognition Challenge (Berg et al., 2010, or just 'the ImageNet Challenge' from now on).

We assume an estimator based on the principle of regularized risk minimization, i.e., we aim to find θ^* such that

$$\theta^* = \operatorname*{argmin}_{\theta} \Bigg[\underbrace{\frac{1}{N} \sum_{n=1}^{N} \Delta(\bar{Y}(x^n;\theta), Y^n)}_{\text{empirical risk}} + \underbrace{\frac{\lambda}{2} \|\theta\|^2}_{\text{regularizer}} \Bigg]. \tag{1}$$

Note that in the case of ImageNet, each image is annotated with a *single* label, while the output space consists of a *set* of K labels; in the other datasets we study, the annotation may consist of any number of labels, including none (we use y to denote a single label, Y to denote a set of labels, and $\mathcal{Y}$ to denote the space of sets of K labels). This setting presents several issues when trying to express (eq. 1) in the framework of large-margin structured prediction: primarily, the margin between the prediction and the groundtruth is not well-defined when they are drawn from different spaces (Tsochantaridis et al., 2005), a problem we discuss in Section 2.3. Perhaps it is for this reason that many of the state-of-the-art methods in the ImageNet Challenge consisted of binary classifiers, such as multiclass SVMs, that merely optimized the score of a single prediction (Lin et al., 2011; Sánchez and Perronnin, 2011).

Table 1 Notation

Notation	Description
x	the feature vector for an image (or just 'an image' for simplicity)
x^n	the feature vector for the n^{th} training image
$\mathcal{X}$	the feature space, i.e., $x^n \in \mathcal{X}$
F	the feature dimensionality, i.e., $F = \lvert x^n \rvert$
N	the total number of training images
y	an image label, consisting of a single object class
Y^n	the set of groundtruth labels for the image x^n
$\mathcal{C}$	the set of classes, i.e., $Y^n \subseteq \mathcal{C}$
C	the total number of classes, i.e., $C = \lvert\mathcal{C}\rvert$
$\bar{Y}(x;\theta)$	the set of output labels predicted by the classifier
$\hat{Y}(x;\theta)$	the output labels resulting in the most violated constraints during column-generation
$\bar{Y}^n$	shorthand for $\bar{Y}(x^n;\theta)$
$\hat{Y}^n$	shorthand for $\hat{Y}(x^n;\theta)$
K	the number of output labels produced by the classifier, i.e., $K = \lvert\bar{Y}^n\rvert = \lvert\hat{Y}^n\rvert$
$\mathcal{Y}$	the space of all possible sets of K labels
θ	a vector parameterizing our classifier
θ^y_{binary}	a binary classifier for the class y
λ	a constant that balances the importance of the empirical risk versus the regularizer
$\phi(x, y)$	the joint parameterization of the image x with the label y
$\Phi(x, Y)$	the joint parameterization of the image x with a *set of labels* Y
$\Delta(Y, Y^n)$	the error induced by the set of labels Y when the correct labels are Y^n
$d(y, y^n)$	a distance measure between the two classes y and y^n in our image taxonomy
Z^n	latent annotation of the image x^n, consisting of $K - \lvert Y^n\rvert$ object classes distinct from Y^n
Ω^n	the 'complete annotation' of the image x^n, i.e., $Y^n \cup Z^n$
G	the number of groundtruth labels used when we analyze the effect of our latent variables

Motivated by the surprisingly good performance of these binary classifiers, in the following sections we shall propose a learning scheme that will 'boost' their performance by re-weighting the dimensions of their parameters so as to take

[1] Note that in McAuley et al. (2011) we assumed that there was only a *single* groundtruth label y^n for each image, as is the case for ImageNet. In the case of the MIR and ImageCLEF datasets there are a variable number (possibly zero) of groundtruth labels for each image, hence the change of notation.

into account the structured nature of the loss function from the ImageNet Challenge.

2.1 The Loss Function

We begin by defining the loss function for the ImageNet Challenge, in which each image is annotated with a single label $Y^n = \{y^n\}$. Each image may contain multiple objects that are not labeled, and the labeled object need not necessarily be the most salient, so a method should not be penalized for predicting 'incorrect' labels in the event that those objects actually appear in the scene. Note that this is not an issue in some similar datasets, such as the Caltech datasets (Griffin et al., 2007), where images have been selected to avoid such ambiguity in the labeling, nor in datasets where all objects are annotated in every image, as in the Pascal Visual Object Challenge (Everingham et al., 2010), or in the MIR and ImageCLEF datasets which we discuss later (Huiskes et al., 2010; Nowak et al., 2011).

To address this issue, a loss is given over a *set* of predicted output labels Y, that only penalizes the method if *none* of those labels is similar to the annotated object. For a training image annotated with a single label $Y^n = \{y^n\}$, the loss incurred by predicting the set of labels Y is given by

$$\Delta(Y, \{y^n\}) = \min_{y \in Y} d(y, y^n). \tag{2}$$

In principle, $d(y, y^n)$ could be any difference measure between the classes y and y^n. If $d(y, y^n) = 1 - \delta(y = y^n)$ (i.e., 0 if $y = y^n$, 1 otherwise), this recovers the ImageNet Challenge's 'flat' error measure. If $d(y, y^n)$ is the shortest-path distance from y^n to the nearest common ancestor of y and y^n in a taxonomic tree, this recovers the 'hierarchical' error measure (which we shall use in our experiments). WordNet is used to build the taxonomic tree for ImageNet, since it is also the source of the object vocabulary (Miller, 1995). Note that this error measure is not symmetric: no penalty is incurred if the prediction y is more specific (with respect to the taxonomy) than the annotation y^n, but a penalty *is* incurred if the prediction is too general.

For problems where multiple groundtruth annotations are available, we desire a loss function that does not penalize the method so long as *any* of the predicted labels are similar to *any* of the groundtruth labels. Using the same difference measure $d(y, y^n)$ as in (eq. 2), our loss becomes

$$\Delta(Y, Y^n) = \min_{y \in Y} \min_{y^n \in Y^n} d(y, y^n) \tag{3}$$

(if there are no training annotations we define $\Delta(Y, \emptyset) = 0$). This is certainly not the only loss function we could choose for multiple labels, but in our case it is motivated by the problem of tag recommendation: we are satisfied so long as *some* plausible tags are suggested to the user. Indeed we could optimize a number of other loss functions using the framework we describe, for example

$$\Delta(Y, Y^n) = \frac{1}{|Y^n|} \sum_{y^n \in Y^n} \min_{y \in Y} d(y, y^n), \tag{4}$$

though for our experiments we use the loss of (eq. 3).

2.2 'Boosting' of Binary Classifiers

Many of the state-of-the-art methods for image classification consist of learning a series of binary 'one vs. all' classifiers that distinguish a single class from all others. That is, for each class $y \in \mathcal{C}$ (where $\mathcal{C}$ is the object vocabulary), one learns a separate parameter vector θ^y_{binary}, and then performs classification by choosing the class with the highest score, using a classifier of the following form:

$$\bar{y}_{\text{binary}}(x) = \operatorname*{argmax}_{y \in \mathcal{C}} \left\langle x, \theta^y_{\text{binary}} \right\rangle. \tag{5}$$

In order to predict a set of K labels, such methods simply return the labels with the K highest scores:

$$\bar{Y}_{\text{binary}}(x) = \operatorname*{argmax}_{Y \in \mathcal{Y}} \sum_{y \in Y} \left\langle x, \theta^y_{\text{binary}} \right\rangle, \tag{6}$$

where $\mathcal{Y}$ is the space of sets of K distinct labels. The above equations describe many of the competitive methods from the ImageNet Challenge, including Lin et al. (2011) and Sánchez and Perronnin (2011).

One obvious improvement is simply to learn a new set of classifiers $\{\theta^y\}_{y \in \mathcal{C}}$ that optimize the structured error measure of (eq. 1). However, given the large number of classes in the ImageNet Challenge ($|\mathcal{C}| = 1000$), and the high dimensionality of standard image features, this would mean simultaneously optimizing several million parameters, which in our experience proved impractical in terms of running time and performance. For the smaller datasets that we study (MIR and ImageCLEF), it is possible to train the individual classifiers directly so as to optimize (eq. 1), though as we shall report doing so does not lead to good performance.

Instead, we would like to leverage the already good classification performance of existing binary classifiers, simply by re-weighting their dimensions to account for the structured nature of (eq. 3). Hence we will learn a *single* parameter vector θ that re-weights the parameters of every class. Our proposed learning framework is designed to extend linear classifiers of the form given in (eq. 6). Given a set of binary classifiers $\{\theta^y_{\text{binary}}\}^{y \in \mathcal{C}}$, we propose a new classifier of the form

$$\bar{Y}(x; \theta) = \operatorname*{argmax}_{Y \in \mathcal{Y}} \sum_{y \in Y} \left\langle x \odot \theta^y_{\text{binary}}, \theta \right\rangle, \tag{7}$$

where $x \odot \theta^y_{\text{binary}}$ is simply the Hadamard product of x (the feature vector) and θ^y_{binary} (the parameter vector). Note that when $\theta = \mathbf{1}$ this recovers precisely the original model of (eq. 6).

To use the standard notation of structured prediction, we define the joint feature vector $\Phi(x, Y)$ as

$$\Phi(x, Y) = \sum_{y \in Y} \phi(x, y) = \sum_{y \in Y} x \odot \theta^{y}_{\text{binary}}, \tag{8}$$

so that (eq. 6) can be expressed as

$$\bar{Y}(x; \theta) = \underset{Y \in \mathcal{Y}}{\operatorname{argmax}} \langle \Phi(x, Y), \theta \rangle \tag{9}$$

(i.e., the predictor is linear in θ). We will use the shorthand $\bar{Y}^n := \bar{Y}(x^n; \theta)$ to avoid excessive notation. In the following sections we shall discuss how structured prediction methods can be used to optimize models of this form.

2.3 The Latent Setting

As mentioned, the joint parameterization of (eq. 8) is problematic, since the energy of the groundtruth labeling Y^n, $\langle \Phi(x^n, Y^n), \theta \rangle$, is not readily comparable with the energy of the predicted output Y, $\langle \Phi(x^n, Y), \theta \rangle$, due to the fact that the size of the two sets is in general different, specifically $|Y^n| \leq |Y|$.

To address this, we propose the introduction of a set of latent variables, $Z = \{Z^1 \dots Z^N\}$, which for each image x^n is designed to encode *the set of objects that appear in x^n that were not annotated*. The full set of labels for the image x^n is now $\Omega^n = Y^n \cup Z^n$ (note that $Y^n \cap Z^n = \emptyset$). If our method outputs K objects, then we fix $|Z^n| = K - |Y^n|$, so that $|\Omega^n| = K$. It is now possible to meaningfully compute the difference between $\Phi(x^n, Y)$ and $\Phi(x^n, \Omega^n)$, where the latter is defined as

$$\Phi(x^n, \Omega^n) = \sum_{y \in Y^n} \phi(x^n, y) + \sum_{z \in Z^n} \phi(x^n, z). \tag{10}$$

The importance of this step shall become clear in Section 3.1, (eq. 15). Note that we still define $\Delta(Y, Y^n)$ only in terms of the training labels Y^n, as in (eq. 3).

Following the programme of Yu and Joachims (2009), learning proceeds by alternately optimizing the latent variables and the parameter vector. Optimizing the parameter vector θ^i given the latent variables Z^i is addressed in Section 3.1; optimizing the latent variables Z^i given the parameter vector θ^{i-1} is addressed in Section 3.2.

3 The Optimization Problem

The optimization problem of (eq. 1) is non-convex. More critically, the loss is a piecewise constant function of θ.[2] A similar problem occurs when one aims to optimize a 0/1 loss in binary classification; in that case, a typical workaround consists of minimizing a surrogate convex loss function that upper-bounds the 0/1 loss, such as the hinge loss, which gives rise to support vector machines. We will now see that we can construct a suitable convex relaxation for the problem defined in (eq. 1).

3.1 Convex Relaxation

Here we use an analogous approach to that of SVMs, notably popularized in Tsochantaridis et al. (2005), which optimizes a convex upper bound on the structured loss of (eq. 1). The resulting optimization problem is

$$[\theta^*, \xi^*] = \underset{\theta, \xi}{\operatorname{argmin}} \left[\frac{1}{N} \sum_{n=1}^{N} \xi_n + \frac{\lambda}{2} \|\theta\|^2 \right] \tag{11a}$$

$$\text{s.t. } \langle \Phi(x^n, \Omega^n), \theta \rangle - \langle \Phi(x^n, Y), \theta \rangle \geq \Delta(Y, Y^n) - \xi_n \tag{11b}$$

$\forall n, Y \in \mathcal{Y}$.

It is easy to see that ξ^*_n upper-bounds $\Delta(\bar{Y}^n, Y^n)$ (and therefore the objective in (eq. 11) upper bounds that of (eq. 1) for the optimal solution). First note that since the constraints of (eq. 11b) hold for all Y, they also hold for $\bar{Y}^n$. Second, the left hand side of the inequality for $Y = \bar{Y}^n$ must be nonpositive since $\bar{Y}(x; \theta) = \operatorname{argmax}_Y \langle \Phi(x, Y), \theta \rangle$. It then follows that $\xi^*_n \geq \Delta(\bar{Y}^n, Y^n)$. This implies that a solution of the relaxation is an upper bound on the solution of the original problem, and therefore the relaxation is well-motivated.

The constraints of (eq. 11b) basically enforce a loss-sensitive margin: θ is learned so that mispredictions Y that incur some loss end up with a score $\langle \Phi(x^n, Y), \theta \rangle$ that is smaller than the score $\langle \Phi(x^n, \Omega^n), \theta \rangle$ of the 'correct' prediction Ω^n by a margin equal to that loss (minus the slack ξ_n). The formulation is a generalization of support vector machines for multi-class problems.

There are two options for solving the convex relaxation of (eq. 11). One is to explicitly include all $N \times |\mathcal{Y}|$ constraints and then solve the resulting quadratic program using one of several existing methods. This may not be feasible if $N \times |\mathcal{Y}|$ is too large. In this case, we can use a constraint generation strategy. This consists of iteratively solving the quadratic program by adding at each iteration the constraint corresponding to the most violated Y for the current model θ and training instance n. This is done by maximizing the violation gap ξ_n, i.e., solving at each iteration the problem

$$\hat{Y}(x^n; \theta) = \underset{Y \in \mathcal{Y}}{\operatorname{argmax}} \{\Delta(Y, Y^n) + \langle \Phi(x^n, Y), \theta \rangle\}, \tag{12}$$

(as before we define $\hat{Y}^n := \hat{Y}(x^n; \theta)$ for brevity). The solution to this optimization problem (known as 'column generation') is somewhat involved, though it turns out to be tractable as we shall see in Section 3.3.

[2] There are countably many values for the loss but uncountably many values for the parameters, so there are large equivalence classes of parameters that correspond to precisely the same loss.

Several publicly available tools implement precisely this constraint generation strategy. A popular example is SvmStruct (Tsochantaridis et al., 2005), though we use BMRM ('Bundle Methods for Risk Minimization'; Teo et al., 2007) in light of its faster convergence properties. Algorithm 1 describes pseudocode for solving the optimization problem of (eq. 11) with BMRM. In order to use BMRM, one needs to compute at the optimal solution ξ_n^* for the most violated constraint $\hat{Y}^n$, both the value of the objective function (eq. 11) and its gradient. At the optimal solution for ξ_n^* with fixed θ we have

$$\langle \Phi(x^n, \Omega^n), \theta \rangle - \langle \Phi(x^n, \hat{Y}^n), \theta \rangle = \Delta(\hat{Y}^n, Y^n) - \xi_n^*. \tag{13}$$

(recall that Ω^n is the 'complete' annotation consisting of the union of the groundtruth and the latent variables). By expressing (eq. 13) as a function of ξ_n^* and substituting into the objective function we obtain the following lower bound on the objective of (eq. 11a):

$$o_i = \frac{1}{N} \sum_n \Delta(\hat{Y}^n, Y^n) - \langle \Phi(x^n, \Omega^n), \theta \rangle + \langle \Phi(x^n, \hat{Y}^n), \theta \rangle + \frac{\lambda}{2} \|\theta\|^2, \tag{14}$$

whose gradient with respect to θ is

$$g_i = \lambda\theta + \frac{1}{N} \sum_n (\Phi(x^n, \hat{Y}^n) - \Phi(x^n, \Omega^n)). \tag{15}$$

The method described above could in principle be used for regularizers other than the ℓ_2 norm (see Teo et al., 2007), though we require that the model is linear in θ (i.e., it can be expressed in the form of (eq. 9)), and that (eq. 12) is tractable. Extensions of such approaches exist, for example kernelized variants are discussed in Yu and Joachims (2008). Here we focus on linear models for efficiency reasons – efficient bundle methods cannot be readily applied to solve the dual problem (see Teo et al. (2007) for details). We refer the reader to Tsochantaridis et al. (2005) and Yu and Joachims (2009) for further discussion of the limitations of this type of approach.

3.2 Learning the Latent Variables

To learn the optimal value of θ, we alternate between optimizing the parameter vector θ^i given the latent variables Z^i, and optimizing the latent variables Z^i given the parameter vector θ^{i-1}. Given a fixed parameter vector θ, the optimal values of the latent variables Z^n can be found greedily; doing so is in fact equivalent to performing inference, with the restriction that the true labels Y^n cannot be part of the latent variable Z^n (see Algorithm 2, Line 5).

It is shown in Yu and Joachims (2009) that this type of alternating optimization is a specific instance of a convex-concave procedure ('CCCP', Yuille and Rangarajan, 2002).

Algorithm 1 Taxonomy Learning

```
 1: Input: training set $\{(x^n, Y^n, Z^n)\}_{n=1}^N$
 2: Output: $\theta$
 3: $\theta := \mathbf{0}$ {in the setting of Algorithm 2, $\theta$ can be 'hot-started' with its
    previous value}
 4: repeat
 5:     for $n \in \{1 \dots N\}$ do
 6:         $\hat{Y}^n := \mathrm{argmax}_{Y \in \mathcal{Y}} \{\Delta(Y, Y^n) + \langle \phi(x^n, Y), \theta \rangle\}$ (see Section 3.3)
 7:     end for
 8:     Compute gradient $g_i$ (equation (eq. 15))
 9:     Compute objective $o_i$ (equation (eq. 14))
10:     $\theta := \mathrm{argmin}_\theta \frac{\lambda}{2} \|\theta\|^2 + \max(0, \max_{j \le i} \left\langle g_j, \theta \right\rangle + o_j)$
11: until converged (see Teo et al. (2007))
12: return $\theta$
```

What this means in practice is that by alternately optimizing θ and Z as in Algorithms 1 and 2, we will arrive at a local optimum of (eq. 1). Naturally this implies that the algorithm is sensitive to initialization, though as we noted, setting $\theta = \mathbf{1}$ recovers the already good performance of our initial binary classifiers, making this an ideal starting point for a local search.

See Yu and Joachims (2009) for further discussion of this type of approach.

Algorithm 2 Taxonomy Learning with Latent Variables

```
1: Input: training set $\{(x^n, Y^n)\}_{n=1}^N$
2: Output: $\theta$
3: $\theta^0 := \mathbf{1}$
4: for $i = 1 \dots I$ {$I$ is the total number of iterations} do
5:     $Z_i^n := \left\{\mathrm{argmax}_{Y \in \mathcal{Y}} \left\langle \Phi(x^n, Y), \theta^{i-1} \right\rangle\right\} \setminus Y^n$ {choose only $K - |Y^n|$
       distinct labels}
6:     $\theta^i :=$ Algorithm1$\left(\left\{\left(x^n, Y^n, Z_i^n\right)\right\}_{n=1}^N\right)$
7: end for
8: return $\theta^I$
```

3.3 Column Generation

Given the loss function of (eq. 3), obtaining the most violated constraints (Algorithm 1, Line 6) takes the form

$$\hat{Y}^n = \operatorname*{argmax}_{Y \in \mathcal{Y}} \left\{ \min_{y \in Y} \min_{y^n \in Y^n} d(y, y^n) + \sum_{y \in Y} \langle \phi(x^n, y), \theta \rangle \right\}, \tag{16}$$

which appears to require enumerating through all $Y \in \mathcal{Y}$, which if there are $C = |\mathcal{C}|$ classes amounts to $\binom{C}{K}$ possibilities. However, if we had an oracle which told us that

$$\operatorname*{argmin}_{y \in \hat{Y}^n} \left[\min_{y^n \in Y^n} d(y, y^n) \right] = c, \tag{17}$$

then (eq. 12) becomes

$$\hat{Y}^n = \operatorname*{argmax}_{Y \in \mathcal{Y}'} \left\{ \min_{y^n \in Y^n} d(c, y^n) + \sum_{y \in Y} \langle \phi(x^n, y), \theta \rangle \right\}, \tag{18}$$

where $\mathcal{Y}'$ is just $\mathcal{Y}$ restricted to those y for which

$$\min_{y^n \in Y^n} d(y, y^n) \geq \min_{y^n \in Y^n} d(c, y^n). \tag{19}$$

The important difference between (eq. 16) and (eq. 18) is simply that the part of (eq. 16) involving the loss has been replaced by a constant, $\min_{y^n \in Y^n} d(c, y^n)$, which we will show is enough to make (eq. 18) tractable. Of course we have yet to determine the value of this constant, though we show below that this can also be done efficiently.[3]

We can obtain the optimal solution to (eq. 18) greedily by sorting $\langle \phi(x^n, y), \theta \rangle$ for each class $y \in \mathcal{C}$ such that

$$\min_{y^n \in Y^n} d(y, y^n) \geq \min_{y^n \in Y^n} d(c, y^n) \tag{20}$$

and simply choosing the top K classes. Since we don't know the optimal value of c in advance, we must consider all $c \in \mathcal{C}$, which means solving (eq. 18) a total of C times (recall that C is the number of classes). Solving (eq. 18) greedily takes $O(C \log C)$ (to sort C values), so that solving (eq. 12) takes $O(C^2 \log C)$.

Although this method works for any loss of the form given in (eq. 3), for the specific distance function $d(y, y^n)$ used for the ImageNet Challenge, further improvements are possible. As mentioned, for the ImageNet Challenge's hierarchical error measure, $d(y, y^n)$ is the shortest-path distance from y^n to the nearest common ancestor of y and y^n in a taxonomic tree. One would expect the depth of such a tree to grow logarithmically in the number of classes, and indeed we find that we always have $d(y, y^n) \in \{0 \ldots 18\}$ (the trees for the other datasets we study are shallower). Since the number of distinct possibilities for $\Delta(Y, Y_n)$ is small, instead of enumerating each possible value of

$$c = \operatorname*{argmin}_{y \in \hat{Y}^n} \left[\min_{y^n \in Y^n} d(y, y^n) \right], \tag{21}$$

we can directly enumerate each value of

$$\delta = \left[\min_{y \in \hat{Y}^n} \min_{y^n \in Y^n} d(y, y^n) \right], \tag{22}$$

i.e., each possible value for the loss $\Delta(Y, Y^n)$. If there are L distinct values of the loss, (eq. 12) can now be solved in $O(LC \log C)$. In ImageNet Challenge we have $L = 19$ whereas $C = 1000$, so this is clearly a significant improvement.

Additional minor improvements can be made, for example we do not need to sort all C values in order to compute the top K items, and we do not need to re-sort all items for each value of the loss δ. The implementation used in our experiments is available online.[4]

4 Experiments

4.1 Binary Classifiers

As previously described, our approach needs, for each class, one binary classifier able to provide some reasonable score as a starting point for the proposed method. Since the objective of this paper is not beating the state-of-the-art, but rather demonstrating the advantage of our structured learning approach to improve overall classification results, we used a standard, simple image classification setup. As mentioned, should the one-vs-all classifiers of Lin et al. (2011) or Sánchez and Perronnin (2011) become available in the future, they should be immediately compatible with the proposed method.

First, images have to be transformed into descriptor vectors sensible for classification using machine learning techniques. For this we have chosen the very popular Bag of Features model (Csurka et al., 2004): dense SIFT features are extracted from each image x^n and quantized using a visual vocabulary of F visual words. Next, the visual words are pooled in a histogram that represents the image. This representation is widely used in state-of-the-art image classification methods, and in spite of its simplicity achieves very good results.

Regarding our basic classifiers, a sensible first choice, considering existing related work, would be to use a Linear SVM for every class. However, since our objective is to predict the correct class of a new image, we would need to compare the raw scores attained by the classifier, which would not be theoretically satisfying. Although it is possible to obtain probabilities from SVM scores using a sigmoid trained with the Platt algorithm, we instead opted to train logistic regressors, which directly give probabilities as outputs and do not depend on a separate validation set.

In order to deal with the computational and memory requirements derived from the large number of training images, we used Stochastic Gradient Descent (SGD) from Bottou and Bousquet (2008) to train the classifiers. SGD is a good choice for our problem, since it has been shown to achieve performance similar to that of batch training methods in a fraction of the time (Perronnin et al., 2010). Furthermore, we validated its performance against that of LibLinear in a small-scale experiment using part of the ImageNet Challenge hierarchy with satisfactory results. One limitation of online learning methods is that the optimization process iterations are limited by the amount of training data available. In order to add more training data, we cycled over all of the training data for 10 epochs.

For the MIR and ImageCLEF experiments we used SIFT features based on the implementation of (van de Sande et al., 2010). Due to the more manageable size of these datasets, the classifiers were trained using LibLinear.

[3] Note that somewhat simpler notation was used in McAuley et al. (2011), in which there was only a *single* output label y^n, but otherwise the idea remains the same.

[4] see http://i.stanford.edu/~julian

Using the above approach, the parameters θ^y_{binary} for each class used in the structured learning methods in the following sections were generated.

4.2 Structured Classifiers on ImageNet Data

For our first experiment, we consider structured classification on the ImageNet Challenge dataset. This dataset consists of 1.35 million images (1.2 million for training, the rest for testing), each of which is labeled with a single positive class.

For every image x^n and every class y we must compute $\langle\phi(x^n, y), \theta\rangle$. Earlier we defined $\phi(x, y) = x \odot \theta^y_{\text{binary}}$. If we have C classes and F features, then this computation can be made efficient by first computing the $(C \times F)$ matrix A whose y^{th} row is given by $(\theta^y_{\text{binary}} \odot \theta)$. Similarly, if we have N images then the set of image features can be thought of as an $(N \times F)$ matrix X. Now the energy of a particular labeling y of x^n under θ is given by the matrix product

$$\langle\phi(x^n, y), \theta\rangle = \left(X \times A^T\right)_{n,y}. \tag{23}$$

This observation is critical if we wish to handle a large number of images and feature vectors of high-dimension. In our experiments, we performed this computation using Nvidia's high-performance BLAS library CUBLAS. Although GPU performance is often limited by a memory bottleneck, this particular application is ideally suited as the matrix X is far larger than either the matrix A, or the resulting product, and X needs to be copied to the GPU only once, after which it is repeatedly reused. After this matrix product is computed, we must sort every row, which can be naïvely parallelized.

In light of these observations, our method is no longer prohibitively constrained by its running time (running ten iterations of Algorithm 2 takes around one day for a single regularization parameter λ). Instead we are constrained by the size of the GPU's onboard memory, meaning that we only used 25% of the training data (half for training, half for validation). To measure the effect of this limitation, we also trained our model using the entire training set, using a parallel implementation on a 64 core machine; here we used Intel's Math Kernel Library to parallelize the matrix multiplication step.[5]

The results of our algorithm using features of dimension $F = 1024$ and $F = 4096$ are shown in Figures 2 and 3, respectively. Note that the 'non-learning' results refer to the version of the algorithm *without* reweighted classifiers, though the initial classifiers are themselves the result of a previous learning stage. Here we ran Algorithm 2 for ten iterations, 'hot-starting' θ^i using the optimal result from the previous iteration. The reduction in training error is also shown for each iteration of Algorithm 2, showing that minimal benefits are gained after ten iterations. We used regularization parameters $\lambda \in \{10^{-1}, 10^{-2} \ldots 10^{-8}\}$, and as usual we report the test error for the value of λ that resulted in the best performance on the validation set. We show the test error for different numbers of nearest-neighbors K, though the method was trained to minimize the error for $K = 5$.

Interestingly, we see negligible benefit when using the entire training set versus using merely 25%. In light of the fact that a single round of training (for the 4096 dimensional features) takes approximately six times longer on the CPU (using non-commodity hardware), we advocate use of the GPU implementation. In the experiments that we consider later, the datasets are sufficiently small that neither memory requirements nor running times present a significant issue.

In both Figures 2 and 3, we find that the optimal θ is non-uniform, indicating that there are interesting relationships that can be learned between the features when a structured setting is used. As hoped, a reduction in test error is obtained over already good classifiers, though the improvement is less significant for the better-performing high-dimensional classifiers.

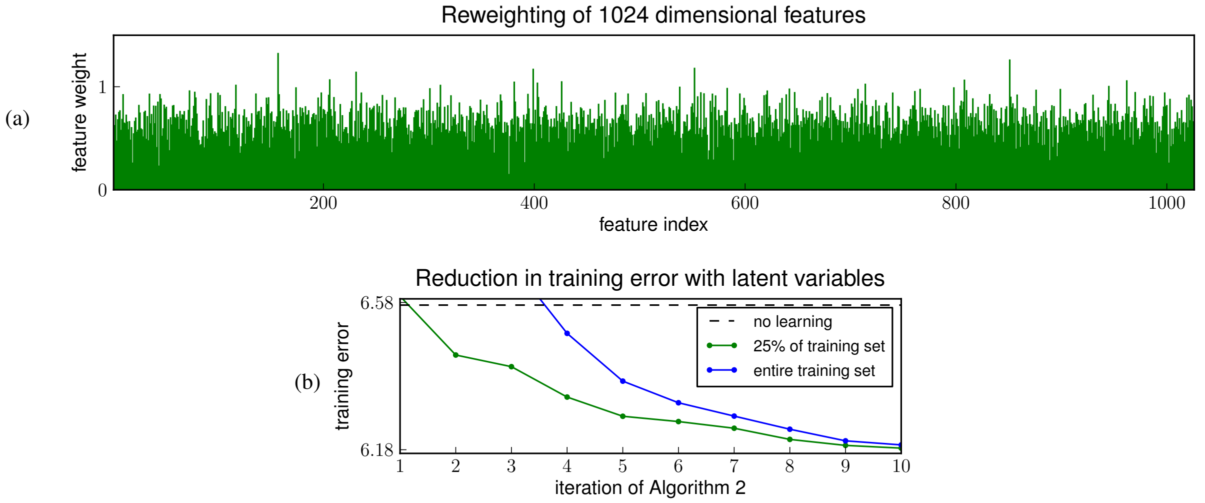
Ideally, we would like to apply our method to features and classifiers like those of Lin et al. (2011) or Sánchez and Perronnin (2011). It remains to be seen whether the setting we have described could yield additional benefits over their already excellent classifiers.

4.3 Multiple Observed Labels

Our model was designed based on the intuition that the latent variables should capture those objects that appear in an image, but are not present in the groundtruth. However, as we observe only a single (positive) label for each image in ImageNet Challenge, we were unable to test this hypothesis on that data. In this experiment, we study two taxonomies for which a complete labeling is provided for each image. By training using only a fraction of the groundtruth labels, this allows us to examine the extent to which the latent variables align with those categories withheld from the groundtruth. These datasets also allow us to measure the benefit obtained by our latent variable model as the labeling becomes 'stronger', i.e., as a more complete groundtruth is provided.

In the *MIR Flickr Retrieval Evaluation* (MIR) dataset, 25,000 images are labeled into 24 object categories (Huiskes et al., 2010). Although the classifiers discussed in Huiskes et al. (2010) are concerned with optimizing average precision, the object categories are organized into topics and subtopics, which can be treated as a taxonomy. The taxonomy we derive is shown in Figure 4 (see Huiskes et al., 2010, Table 2). Although the small number of categories in this

[5] http://software.intel.com/en-us/articles/intel-mkl/



(c)

Test error:

	1nn	2	3	4	5
Before learning	11.35	9.29	8.08	7.25	6.64
After learning (25% of training set)	10.88	8.85	7.71	6.93	6.36
After learning (entire training set)	10.81	8.82	7.68	6.94	6.37

Fig. 2 Results for training with 1024 dimensional features on ImageNet Challenge data. (a) feature weights; (b) reduction in training error during each iteration of Algorithm 2; (c) error for different numbers of nearest-neighbors K (the method was trained to optimize the error for $K = 5$). Results are reported for the best value of λ on the validation set (here $\lambda = 10^{-4}$). Due to the large size of the datasets in question, standard errors are $\simeq 0$ for all datapoints.

dataset does not demand the use of a hierarchical loss (the labels are sufficiently distinct that the $0/1$ loss is sufficient), it is valuable as a first step in identifying the function of the latent variables in our model.

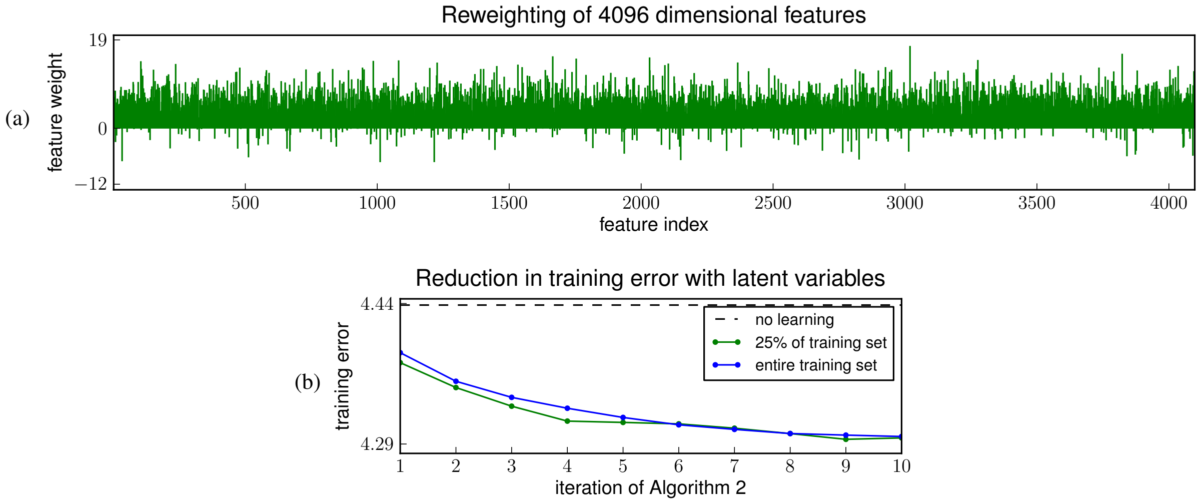
The *ImageCLEF* dataset uses a subset of 18,000 images from the MIR dataset (Nowak and Huiskes, 2010). The images are categorized into 99 concepts, forming a richer and deeper hierarchy than that of the MIR dataset. The taxonomy for the ImageCLEF dataset is shown in Figure 5.

In both of these datasets, images may have any number of labels from the label vocabulary, including none (the MIR and ImageCLEF datasets have up to 14 and 26 labels per image, respectively). To analyze the role of the latent variables, we randomly select a fixed number of (up to) G labels for each image. These G labels form our training annotations Y^n. Note that when $G = 1$ we recover precisely the setting used in the experiment of Section 4.2 for the ImageNet Challenge data.

Results for learning on both the MIR and the ImageCLEF datasets for different values of G are shown in Figure 8. Two aspects of these results are interesting at first glance: firstly, the improvement of learning over non-learning is the most significant in the MIR taxonomy, in spite of our previous comment that a loss derived from this taxonomy, being the shallowest, most closely resembles the simpler $0/1$ loss. In fact, on further inspection we discover that among all of our experiments, the largest improvements are achieved in the *smallest* taxonomies. However, it should be noted that the loss of (eq. 3) still differs from the $0/1$ loss due to the fact that we predict multiple labels simultaneously, so this finding merely reveals that our model is better able to leverage the structured nature of the loss when the label vocabulary is smaller.

Figure 6 shows the results on some example images from ImageCLEF, for $G = 1$ (i.e., we randomly choose a single label from the complete groundtruth for training). Some common patterns emerge which explain how our latent variable model is able to leverage the structured nature of the loss function: Firstly, higher scores are given to extremely common classes such as ‘no blur’, which are rarely given high scores by the original binary classifiers. Secondly, the original classifiers often predict semantically similar labels (such as ‘happy’ and ‘funny’, or ‘shadow’ and ‘night’), whereas more semantically distinct classes are chosen after learning.

Secondly, we note that as G increases (meaning that the groundtruth becomes more complete), the relative improvement of learning over non-learning *increases* in almost all cases (naturally the *absolute* value of the loss decreases for larger G, as the problem becomes easier due to the ‘min’ in (eq. 3)). This is an interesting result, since for images with fully-labeled groundtruth, the latent variables no longer play any readily interpretable role. Again we note that no matter what the values of the latent variables, we still gain a considerable advantage in all cases simply due to the fact that we



Test error:

(c)

	1nn	2	3	4	5
Before learning	9.27	7.29	6.23	5.53	5.03
After learning (25% of training set)	9.02	7.08	6.05	5.38	4.91
After learning (entire training set)	9.00	7.06	6.04	5.37	4.90

Fig. 3 Results for training with 4096 dimensional features on ImageNet Challenge data. (a) feature weights; (b) reduction in training error during each iteration of Algorithm 2; (c) error for different numbers of nearest-neighbors K (the method was trained to optimize the error for $K = 5$). Results are reported for the best value of λ on the validation set (here $\lambda = 10^{-6}$).

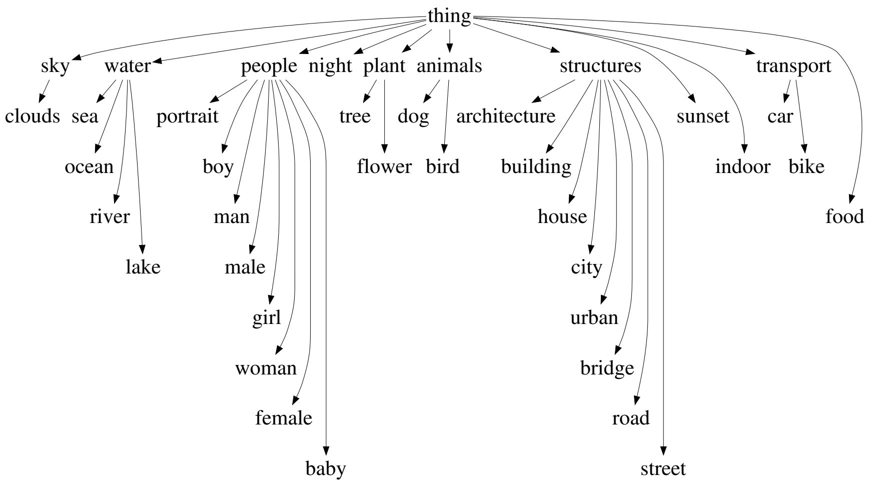


Fig. 4 The MIR taxonomy. Twenty-four of the tree's nodes are the possible labels for each image.

are optimizing the correct loss. Ultimately, what this does imply is that while the 0/1 loss (or some surrogate such as the hinge loss) appears to be a reasonable proxy for the loss of (eq. 3) when $G = 1$, the importance of optimizing the correct loss becomes more important when it is a function of multiple groundtruth labels.

In Figure 9 we assess whether the predicted values of the latent variables Z^n match those groundtruth annotations that were withheld from Y^n. If we have C classes and $G' = \min(G, |Y^n|)$ groundtruth labels (i.e., we have G labels except when the complete annotation Y^n has fewer than G labels), then the number of ways of predicting the remaining $K - P$ labels is

$$\binom{C - G'}{K - G'}. \tag{24}$$

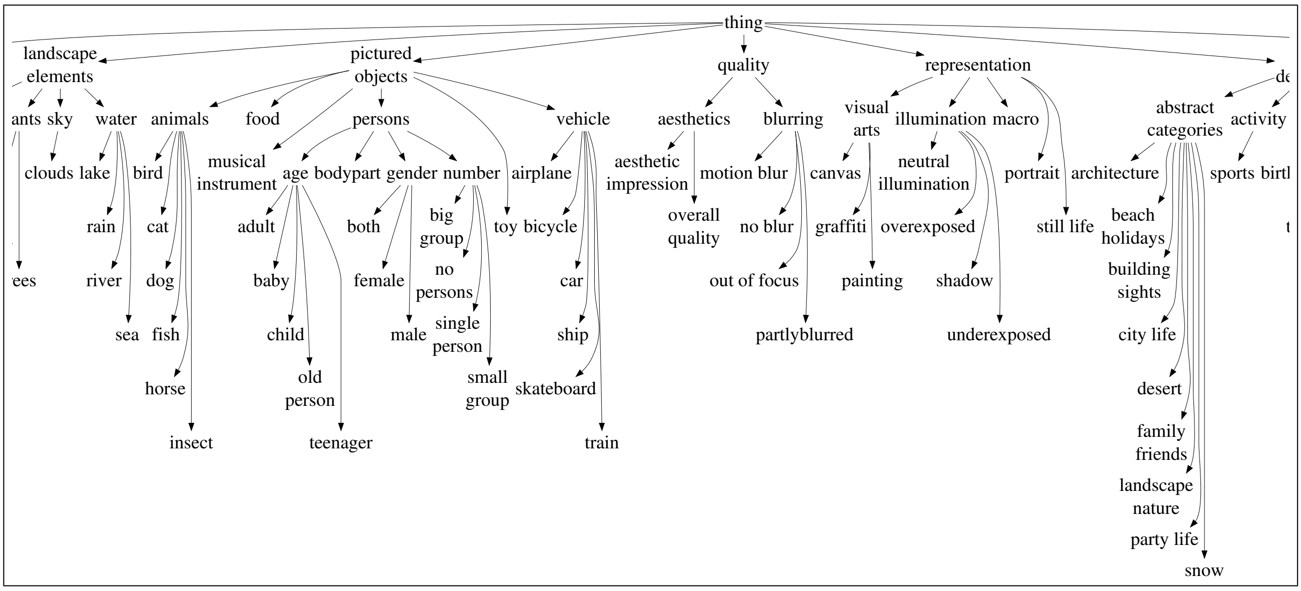


Fig. 5 A partial view of the ImageCLEF taxonomy. The possible labels for each image are the ninety-nine leaves of the tree.

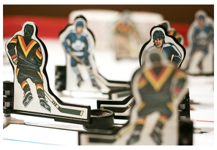
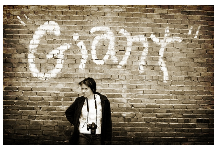
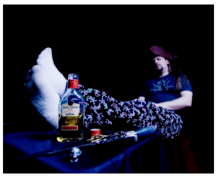
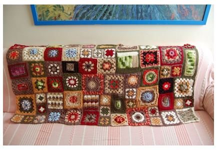
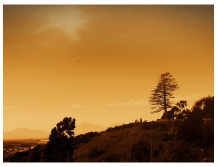
		predicted labels		loss	
image	‘correct’ label	before learning	after learning	before	after
	no persons	partly blurred, happy, funny, neutral illumination, birthday	no persons, partly blurred, active, happy, still life	4	0
	single person	outdoor, no blur, day, sports, water	no persons, no blur, day, outdoor, visual arts	4	1
	single person	night, shadow, underexposed, artificial, scary	no blur, no persons, adult, male, visual arts	4	1
	day	no blur, neutral illumination, no persons, big group, child	no persons, no blur, day, summer, cute	3	0
	no persons	sky, clouds, outdoor, day, mountains	no persons, sky, outdoor, day, no blur	4	0

Fig. 6 Example results on some images from ImageCLEF. Recall that as the ‘correct’ labels we randomly choose a single label from the complete groundtruth. Note that common classes such as ‘no blur’ are predicted more frequently in the hierarchical model, and that semantically similar labels (such as ‘happy’ and ‘funny’, or ‘shadow’ and ‘night’), are rarely predicted together.

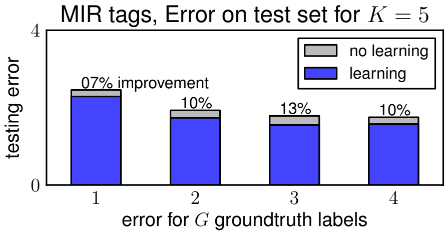


Fig. 7 Improvement of learning over non-learning for the taxonomy defined over MIR tags. All results are reported for the best value of λ on the validation set.

The number of ways that c of them can be 'correct' (i.e., they appear in the withheld groundtruth) is

$$\underbrace{\binom{K-G'}{c}}_{\text{correct predictions}} \times \underbrace{\binom{C-G'-c}{K-G'-c}}_{\text{incorrect predictions}}. \tag{25}$$

Thus the expected number of correct predictions made by a random classifier is

$$\frac{\sum_{c=1}^{K-G'} c \times \binom{K-G'}{c} \times \binom{C-G'-c}{K-G'-c}}{\binom{C-G'}{K-G'}}. \tag{26}$$

During each iteration of Algorithm 2, we measure the fraction of withheld groundtruth labels that appear in the latent variable Z_n for each image, normalized by (eq. 26). When $G = 1$ (as is the case in ImageNet Challenge), we see that for both the MIR and ImageCLEF datasets, the latent variables gradually align with the unannotated objects, as we had expected. However, on the MIR dataset, for all $G > 1$, we see that after an initial period of alignment, the latent variables gradually become *less* similar to the withheld groundtruth; apparently the model is better able to leverage the latent variables by assigning them some role other than matching the withheld groundtruth. On the ImageCLEF dataset, for $G < 4$, the latent variables actually agree with the groundtruth *less* than would be expected of a random classifier, implying that the latent variables play some other role altogether.

Although it is indeed somewhat difficult to interpret the complex dynamics of the upper 8 plots in Figure 9, an intuitive picture emerges after a sufficient number of iterations. The two plots in the bottom of Figure 9 show at the tenth iteration, the agreement between the latent variables and withheld groundtruth increases with G (the number of training labels). We observe a monotonic improvement, which indicates that the more labels we know, the better we can predict the missing labels. This is intuitive in the sense that it agrees with the fact that we use a structured loss that accounts for dependencies between the predicted labels.

4.4 Image Tagging in a Taxonomy

Also provided in the MIR dataset are the Flickr tags for each image. The loss function of (eq. 3) is a natural one for the problem of tag recommendation, as one is satisfied so long as *some* reasonable tags are suggested to the user. It is also natural to penalize 'incorrect' tags using a taxonomy, as the space of all possible tags is too large for the 0/1 loss to be practical (in the 25,000 images in the MIR dataset, there are around 20,000 unique tags). Such a dataset differs from those of Sections 4.2 and 4.3: while the groundtruth may contain multiple labels per image (as with MIR and ImageCLEF), it may still be incomplete (as with ImageNet Challenge), in the sense that we are never certain whether an image *couldn't* have been assigned a particular tag.

The problem of automatically deriving a taxonomy from image tags is studied in Setia and Burkhardt (2007), though for the current experiment it is simpler to manually assign the most commonly used Flickr tags to existing concepts in WordNet (Miller, 1995).

Among the 200 most popular tags in the MIR dataset, 152 correspond to readily identifiable concepts in WordNet (of the other 48, many are camera brands or non-English words). Having identified corresponding concepts in WordNet, we can define our loss function much as is done for the ImageNet Challenge, i.e., by taking the shortest path distance in the WordNet taxonomy from the correct tag to the nearest common ancestor of the predicted and the correct tag. On average, each of the 25,000 images contains 1.44 of these 152 tags. A selection of the tags, and their relationships via homonymns in WordNet are shown in Figure 10.

Results for learning on the MIR tag taxonomy are shown in Figures 7 and 11. In Figure 7 we use the experimental setup from Section 4.3, i.e., we withhold some fraction of the groundtruth labels so as to analyze the effect of training with 'stronger' groundtruth information; results are similar to those reported in Section 4.3. In Figure 11 we use the experimental setup from Section 4.2, i.e., we simply use *all* of the available training evidence. Here we predict $K = 10$ nearest-neighbors, which in practice could represent suggesting ten tags to a user in a tag-recommendation system. Learning achieves an improvement over non-learning of approximately 12% when predicting ten tags, a more substantial improvement than what we observed for ImageNet Challenge.

4.5 Optimization of Flat Losses

In order to optimize the performance measure used in the ImageNet Challenge, we had to account for both the hierarchical nature of the loss, as well as the fact that multiple predictions can be made simultaneously. Although we have demonstrated the benefit of optimizing such performance

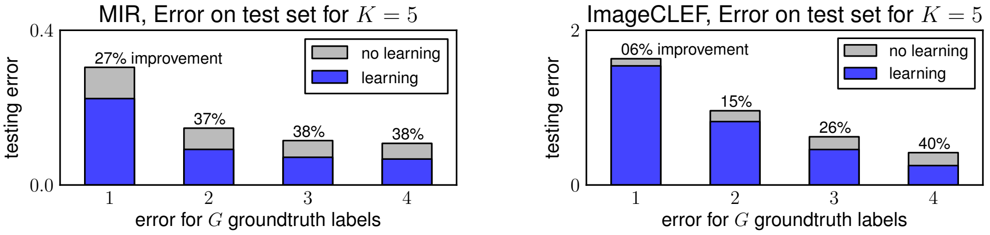


Fig. 8 Improvement of learning over non-learning for the MIR and ImageCLEF taxonomies as the number of groundtruth labels G increases. 'Testing error' refers to the average loss across all images in the test set. The plots are annotated to show the percentage improvement of learning over non-learning. All results are reported for the best value of λ on the validation set. Note that the final two bars for the MIR dataset are almost identical simply due to the fact that few images have as many as four labels owing to the small size of the label vocabulary.

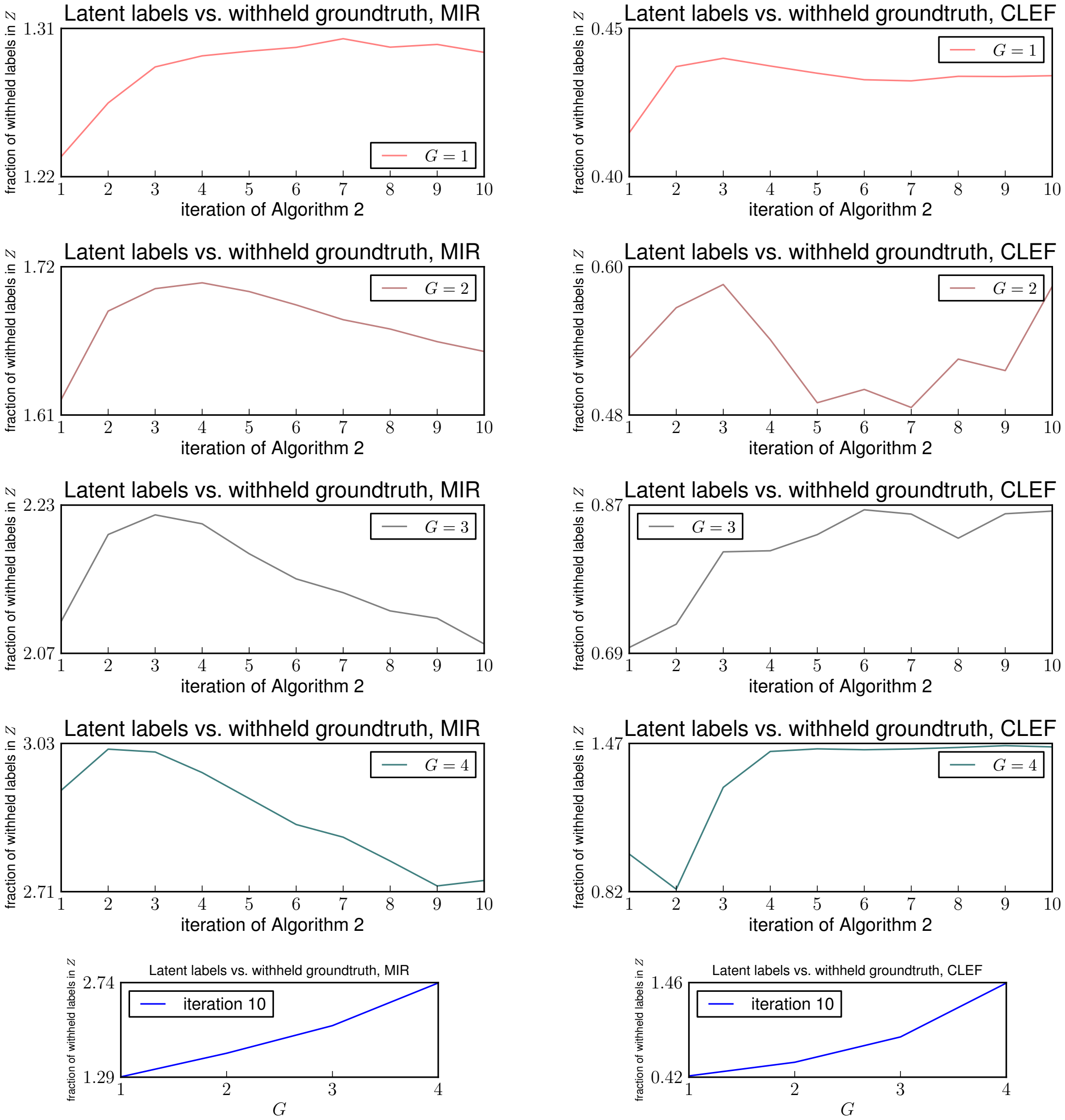


Fig. 9 The above plots measure how closely the latent variables match those labels withheld from the groundtruth during training. The y-axis measures the fraction of withheld groundtruth labels that appear in the latent indicator Z^i (normalized by the expected number of correct predictions made by a random classifier). The eight upper plots show this quantity as a function of the iteration of Algorithm 2, while the two bottom plots show this quantity for iteration 10, as a function of the number of known labels G. The two bottom plots reveal that the level of agreement between the predicted latent variables and the withheld groundtruth increases monotonically in G, which is expected since our structured loss models dependencies between the predicted labels.

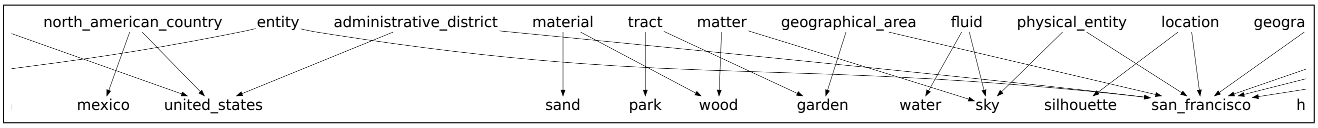


Fig. 10 A partial view of the taxonomy defined on MIR tags. The nodes on the bottom level are (the concepts corresponding to) the tags observed in the MIR dataset, while the nodes on the top level are the common ancestors via which they are related using our loss function (i.e., each edge represents a *path* in the WordNet hierarchy, which may have length greater than one).

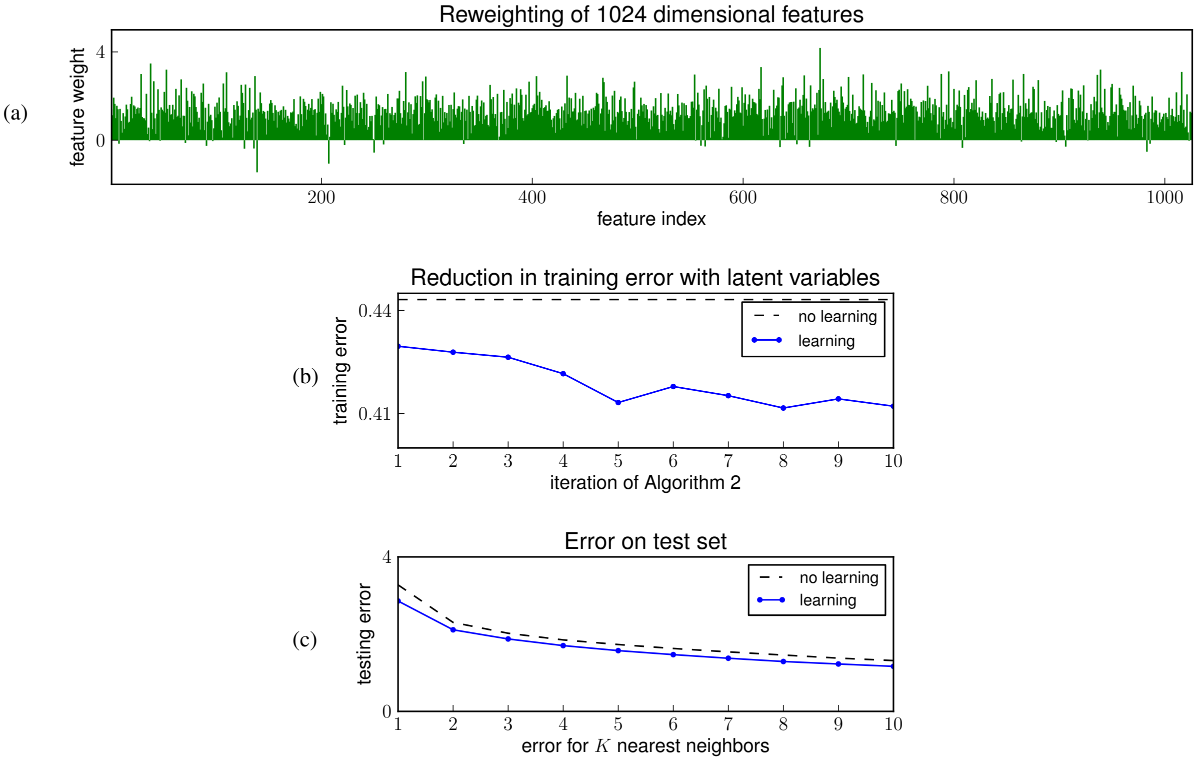


Fig. 11 Results for training with 1024 dimensional features on MIR tag data. (a) feature weights; (b) reduction in training error during each iteration of Algorithm 2; (c) error for different numbers of nearest-neighbors K (the method was trained to optimize the error for $K = 10$). Results are reported for the best value of λ on the validation set (here $\lambda = 10^{-5}$). Note that although Algorithm 2 should produce a monotonic decrease in training error (see Yuille and Rangarajan, 2002), we occasionally observe increase in training error when Algorithm 1 fails to converge.

measures directly, it is unclear to which of these two aspects the improvement owes. One way to test this is to see whether we can achieve similar gains using our latent variable model using 'flat' (i.e., 0/1) losses.

Recall that the error measure used to evaluate performance in the ImageNet Challenge took the form

$$\Delta(Y, \{y^n\}) = \min_{y \in Y} d(y, y^n), \tag{27}$$

where $d(y, y^n)$ was some difference measure between a predicted label y and the groundtruth label y^n. So far we have assumed that $d(y, y^n)$ measured the shortest-path distance from y^n to the nearest common ancestor of y and y^n in a taxonomic tree, corresponding to the ImageNet Challenge's 'hierarchical' performance measure. The ImageNet Challenge's 'flat' performance measure replaces this by

$$d(y, y^n) = 1 - \delta(y = y^n) \tag{28}$$

(i.e., 0 if $y = y^n$, 1 otherwise). In the context of (eq. 27) this means that a loss of 0 is achieved so long as the groundtruth label y^n appears in Y, and 1 otherwise.

Optimizing the ImageNet Challenge's 'flat' performance measure using our latent variable model simply means replacing the hierarchical difference measure used in the previous experiments by that of (eq. 28). Results on the ImageNet, CLEF, and MIR datasets, using 1024 dimensional features are shown in Figure 12. The results are similar (in terms of percentage improvement) to what we reported in Sections 4.2 and 4.3 (except for ImageNet, where the improvement is smaller). Again the error is smallest on the MIR dataset simply due to the fact that it has the smallest label vocabulary.

To study the converse problem of optimizing the hierarchical performance measure *without* introducing latent vari-

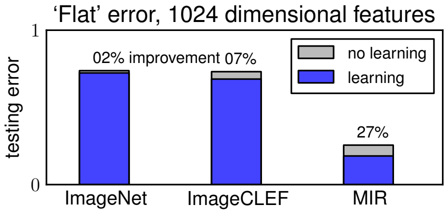


Fig. 12 Improvement of learning over non-learning using the ImageNet Challenge's 'flat' evaluation measure on all three datasets. All results are reported for the best value of λ on the validation set.

ables, we use the hierarchical difference measure from the previous experiments, but optimize the performance of only a single prediction. As in the previous experiments, we evaluate the method using the hierarchical performance measure from ImageNet. We found that doing so did not improve over the non-learning performance (i.e., the performance of the original binary predictors). This is not surprising, since we are no longer optimizing the same performance measure on which the method is evaluated.

It should be noted that Cai and Hofmann (2004) successfully reported the benefit of directly optimizing hierarchical performance measures for multiclass document classification problems, and Binder et al. (2011) reported similarly promising results for problems from computer vision. However, their experiments differ critically from what we present here due to the fact that their datasets have a small label vocabulary and typically only a single plausible label per document (meaning that they can directly evaluate their algorithms on a single prediction).

4.6 Single-Stage Learning

In the case of ImageNet, we made the argument that it is not feasible to learn parameters for all classes simultanetously in a single unified stage, due to the high-dimensionality of the parameters involved (e.g. learning all parameters for the classifiers in 4.2 would require optimizing millions of parameters simultaneously). On the other hand, MIR and ImageCLEF have only 24 and 99 categories (respectively), so that learning all parameters simultaneously ought to be feasible, at least with respect to running time.

Recall that in (eq. 7) we assumed a classifier of the form

$$\bar{Y}(x;\theta) = \operatorname*{argmax}_{Y \in \mathcal{Y}} \sum_{y \in Y} \left\langle x \odot \theta^{y}_{\text{binary}}, \theta \right\rangle, \tag{29}$$

where $\theta^{y}_{\text{binary}}$ was assumed to be given as input from a previous learning stage, so that our algorithm only had to learn the weighting factor θ. A single stage approach replaces this classifier by one of the form

$$\bar{Y}'(x;\theta) = \operatorname*{argmax}_{Y \in \mathcal{Y}} \sum_{y \in Y} \left\langle x, \theta^{y}_{\text{binary}} \right\rangle, \tag{30}$$

where each $\theta^{y}_{\text{binary}}$ is a parameter vector to be learned. Note that this model is linear in the concatenation of all model parameters, $(\theta^{1}_{\text{binary}}, \dots, \theta^{C}_{\text{binary}})$, meaning that it is amenable to the same structured learning approaches we described in Section 3.

In principle the model of (eq. 30) is a generalization of the original model of (eq. 7), though we found that training the model of (eq. 30) led to inferior performance on both MIR and ImageCLEF data (in fact the performance was inferior to the one-vs-all classifiers described in Section 4.1). There are several possible explanations for this phenomenon: Firstly, given that the number of parameters exceeds the number of training images, overfitting is certainly a problem that may be alleviated by a two-stage approach; even for ImageNet, where there are over one million images for training, this problem would persist due to the high number of classes. Secondly, in a two-stage approach the base classifiers $\theta^{y}_{\text{binary}}$ are trained using methods that differ significantly from the max-margin objective of (eq. 11a).

In terms of running time the model of (eq. 30) was also inferior to that of (eq. 7), leading to an approximately 20-fold running time increase in the case of MIR, and a 50-fold increase in the case of ImageCLEF. The same experiment on the ImageNet Challenge data proved impractical in terms of running time.

5 Discussion

The proposed model leads to a reduction in error for all of the hierarchical labeling problems we considered in Section 4. The fact that we achieve the largest benefits in the smallest hierarchies is possibly explained by the findings of Deng et al. (2009), who show that in large hierarchies, confusion tends to occur between semantically similar classes, even when optimizing a 0/1 loss. Alternately, for small hierarchies the classes are more semantically distinct, so that optimizing the hierarchical loss changes the predictions significantly. The fact that we achieve the largest benefits when we have additional groundtruth labels simply reflects the fact that the loss we are optimizing is structured and takes into account label dependencies through the hierarchy.

It is interesting to discover that the *smallest* benefit is obtained when we have a large label vocabulary and only a single groundtruth label, as is the case for the ImageNet Challenge. This may be an indication that the hierarchical information is not informative in such cases, so that the 0/1 loss becomes a good proxy for the hierarchical loss in question. The fact that the best performing methods from the ImageNet Challenge competition in terms of the 0/1 loss were also the best performing in terms of the hierarchical loss provides weak evidence for this assertion. Alternately, it may simply reflect the fact that this is the version of the problem that has received the most attention, so that the state-of-the-art binary classifiers are sufficiently accurate as to be able to overcome the weakness of using the incorrect loss.

Image tags derived from image hosting websites like Flickr are a natural source of weakly-labeled data from large vocabularies, where hierarchical losses are sensible since many tags corresponding to semantically similar concepts can be used interchangeably. Indeed we discovered that most of the commonly used tags in Flickr correspond to readily indentifiable concepts from WordNet, obviating the need to learn a hierarchy directly from the data. However, an obvious danger of using a hierarchial loss is that certain types of tags, while semantically similar, are not interchangable. For instance, while 'San Francisco' and 'New York City' are visually and semantically close (in terms of the WordNet hierarchy), a tag recommendation system that suggests one in place of the other would not be useful. Optimizing for error measures that are sensitive to this fact remains an avenue for future work.

One key limitation of our formulation is that we apply the *same* re-weighting for the parameter vectors of *all* categories. This assumption makes optimization more tractable and leads to an efficient solution. However this is suboptimal in the sense that we would like to have re-weightings that are to some extent dedicated to different categories, while at the same time avoiding the need to have multiple independent re-weighing vectors for each category. This would possibly suggest a strategy that shares parameters across different categories, in the spirit of Lampert et al. (2009). Obtaining a scalable algorithm in this setting however would be a challenge, and is left as future work.

6 Conclusion

Large-scale, collaboratively labeled image datasets embedded in a taxonomy naturally invite the use of robust, structured losses, in the sense that they should account for both the hierarchical nature of the taxonomy and for the inconsistencies in the labeling process. However, on datasets such as ImageNet, the state-of-the-art methods still use one-vs-all classifiers, which do not account for the structured nature of such losses, nor for the imperfect nature of the annotation. We have outlined the computational challenges involved in using structured methods, shedding some light on why they have not been used before in this task. By exploiting a number of computational tricks, and by using recent advances on structured learning with latent variables, we have been able to formulate learning in this task as the optimization of a loss that is both structured and robust to weak labeling. Better yet, our method leverages existing one-vs-all classifiers, essentially by re-weighting, or 'boosting' their dimensions to directly account for the structured nature of the loss. In practice this leads to improvements in the hierarchical loss of already good one-vs-all classifiers.

Acknowledgements

Part of this work was carried out while both Arnau Ramisa and Tibério Caetano were at INRIA Grenoble, Rhône-Alpes, and while Julian McAuley was at NICTA and the ANU. NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program. The work was partially funded by the QUAERO project supported by OSEO, the French State agency for innovation and MICINN under the project MIPRCV Consolider Ingenio CSD2007-00018.

References

Evgenly Bart, Ian Porteous, Pietro Perona, and Max Welling. Unsupervised learning of visual taxonomies. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008. 4

Alex Berg, Jia Deng, and Fei-Fei Li. ImageNet large scale visual recognition challenge 2010. http://www.image-net.org/challenges/LSVRC/2010/index, 2010. 1, 2, 5

Alexander Binder, Klaus-Robert Mller, and Motoaki Kawanabe. On taxonomies for multi-class image categorization. *International Journal of Computer Vision*, page 121, 2011. 3, 4, 17

Matthew Blaschko, Andrea Vedaldi, and Andrew Zisserman. Simultaneous object detection and ranking with weak supervision. In *Advances in Neural Information Processing Systems*, 2010. 4

Léon Bottou and Olivier Bousquet. The tradeoffs of large scale learning. In *Advances in Neural Information Processing Systems*, 2008. 9

Serhat S. Bucak, Rong Jin, and Anil K. Jain. Multi-label learning with incomplete class assignments. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2011. 4

Lijuan Cai and Thomas Hofmann. Hierarchical document categorization with support vector machines. In *Conference on Information and Knowledge Management*, 2004. 3, 17

Gabriela Csurka, Christopher R. Dance, Lixin Fan, Jutta Willamowski, and Cédric Bray. Visual categorization with bags of keypoints. In *ECCV Workshop on Statistical Learning in Computer Vision*, 2004. 9

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009. 1, 2, 17

Jia Deng, Alexander C. Berg, Kai Li, and Li Fei-Fei. What does classifying more than 10,000 image categories tell us? In *European Conference on Computer Vision*, 2010. 2

Jia Deng, Alexander C. Berg, and Li Fei-Fei. Hierarchical semantic indexing for large scale image retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2011. 3

Thomas Deselaers and Vittorio Ferrari. Visual and semantic similarity in imagenet. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2011. 3

Ivica Dimitrovski, Dragi Kocev, Suzana Loskovska, and Saso Džeroski. Detection of visual concepts and annotation of images using ensembles of trees for hierarchical multi-label classification. *Recognizing Patterns in Signals, Speech, Images and Videos*, pages 152–161, 2010. 4

Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The Pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010. 6

Gregory Griffin, Alex Holub, and Pietro Perona. Caltech-256 object category dataset. Technical Report 7694, California Institute of Technology, 2007. 6

Matthieu Guillaumin, Thomas Mensink, Jakob Verbeek, and Cordelia Schmid. TagProp: Discriminative metric learning in nearest neighbor models for image auto-annotation. In *International Conference on Computer Vision*, 2009. 4

Matthieu Guillaumin, Jakob Verbeek, and Cordelia Schmid. Multimodal semi-supervised learning for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2010. 4, 5

Mark J. Huiskes and Michael S. Lew. The MIR Flickr retrieval evaluation. In *International Conference on Multimedia Information Retrieval*, 2008. 2

Mark J. Huiskes, Bart Thomee, and Michael S. Lew. New trends and ideas in visual concept detection: the MIR Flickr retrieval evaluation initiative. In *International Conference on Multimedia Information Retrieval*, 2010. 2, 6, 10

Herve Jégou, Matthijs Douze, and Cordelia Schmid. Product quantization for nearest neighbor search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33 (1):117–128, 2010. 3

Motoaki Kawanabe, Alexander Binder, Christina Muller, and Wojciech Wojcikiewicz. Multi-modal visual concept classification of images via Markov random walk over tags. In *IEEE Workshop on Applications of Computer Vision*, 2011. 5

Byung Soo Kim, Jae Young Park, Anush Mohan, Anna Gilbert, and Silvio Savarese. Hierarchical classification of images by sparse approximation. In *British Machine Vision Conference*, 2011. 3

Christoph H. Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009. 18

Victor Lavrenko, R. Manmatha, and Jiwoon Jeon. A model for learning the semantics of pictures. In *Advances in Neural Information Processing Systems*, 2003. 4

Yuanqing Lin, Fengjun Lv, Shenghuo Zhu, Ming Yang, Timothee Cour, and Kai Yu. Large-scale image classification: fast feature extraction and SVM training. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2011. 1, 2, 5, 6, 9, 10

Marcin Marszałek and Cordelia Schmid. Constructing category hierarchies for visual recognition. In *European Conference in Computer Vision*, 2008. 4

Julian McAuley, Arnau Ramisa, and Tibério Caetano. Optimization of robust loss functions for weakly-labeled image taxonomies: An ImageNet case study. In *Energy Minimization Methods in Computer Vision and Pattern Recognition*, 2011. 2, 5, 9

Thomas Mensink, Jakob Verbeek, and Gabriela Csurka. Learning structured prediction models for interactive image labeling. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2011. 4

George A. Miller. WordNet: A lexical database for English. *Communications of the ACM*, 38:39–41, 1995. 6, 14

Sean Moran and Victor Lavrenko. Optimal tag sets for automatic image annotation. In *British Machine Vision Conference*, 2011. 4

Stefanie Nowak and Mark J. Huiskes. New strategies for image annotation: Overview of the photo annotation task at ImageCLEF 2010. In *CLEF (Notebook Papers/LABs/Workshops)*, 2010. 11

Stefanie Nowak, Karolin Nagel, and Judith Liebetrau. The CLEF 2011 photo annotation and concept-based retrieval tasks. *Working Notes of CLEF*, 2011. 2, 6

Florent Perronnin, Jorge Sánchez, and Thomas Mensink. Improving the Fisher kernel for large-scale image classification. *European Conference on Computer Vision*, 2010. 2, 9

Olga Russakovsky and L. Fei-Fei. Attribute learning in large-scale datasets. In *ECCV Workshop on Parts and Attributes*, 2010. 3

Jorge Sánchez and Florent Perronnin. High-Dimensional Signature Compression for Large-Scale Image Classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2011. 1, 3, 5, 6, 9, 10

Lokesh Setia and Hans Burkhardt. Learning Taxonomies in Large Image Databases. In *ACM SIGIR Workshop on*

Multimedia Information Retrieval, 2007. 14

Qinfeng Shi, James Petterson, Gideon Dror, John Langford, Alex Smola, Alex Strehl, and Vishy Vishwanathan. Hash Kernels. In *Artificial Intelligence and Statistics*, 2009. 3

Josef Sivic, Brian C. Russell, Andrew Zisserman, William T. Freeman, and Alexei A. Efros. Unsupervised discovery of visual object class hierarchies. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008. 4

Choon Hui Teo, Alex Smola, S. V.N. Vishwanathan, and Quoc Viet Le. A scalable modular convex solver for regularized risk minimization. In *Knowledge Discovery and Data Mining*, 2007. 8

Antonio Torralba, Rob Fergus, and William T. Freeman. 80 million tiny images: a large data set for nonparametric object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(11):1958–70, 2008. 2

Ioannis Tsochantaridis, Thorsten Joachims, Thomas Hofmann, and Yasemin Altun. Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, 6:1453–1484, 2005. 5, 7, 8

Koen E. A. van de Sande, Theo Gevers, and Cees G. M. Snoek. Evaluating color descriptors for object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1582–1596, 2010. 9

Jakob Verbeek, Matthieu Guillaumin, Thomas Mensink, and Cordelia Schmid. Image annotation with TagProp on the MIR Flickr set. In *International Conference on Multimedia Information Retrieval*, 2010. 4

Hua Wang, Heng Huang, and Chris Ding. Image annotation using bi-relational graph of images and semantic labels. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2011. 4

Chun-Nam Yu and Thorsten Joachims. Training structural svms with kernels using sampled cuts. In *Knowledge Discovery and Data Mining*, 2008. 8

Chun-Nam Yu and Thorsten Joachims. Learning structural SVMs with latent variables. In *International Conference on Machine Learning*, 2009. 2, 7, 8

Alan Yuille and Anand Rangarajan. The concave-convex procedure (CCCP). In *Advances in Neural Information Processing Systems*, 2002. 8, 16