Learning structured prediction models for interactive image labeling

Thomas Mensink, Jakob Verbeek, Gabriela Csurka

Learning Structured Prediction Models for Interactive Image Labeling

Thomas Mensink[1,2] Jakob Verbeek[2] Gabriela Csurka[1]

[1]Xerox Research Centre Europe [2]LEAR - INRIA Rhône-Alpes

firstname.lastname@xrce.xerox.com firstname.lastname@inrialpes.fr

Abstract

We propose structured models for image labeling that take into account the dependencies among the image labels explicitly. These models are more expressive than independent label predictors, and lead to more accurate predictions. While the improvement is modest for fully-automatic image labeling, the gain is significant in an interactive scenario where a user provides the value of some of the image labels. Such an interactive scenario offers an interesting trade-off between accuracy and manual labeling effort. The structured models are used to decide which labels should be set by the user, and transfer the user input to more accurate predictions on other image labels. We also apply our models to attribute-based image classification, where attribute predictions of a test image are mapped to class probabilities by means of a given attribute-class mapping. In this case the structured models are built at the attribute level. We also consider an interactive system where the system asks a user to set some of the attribute values in order to maximally improve class prediction performance. Experimental results on three publicly available benchmark data sets show that in all scenarios our structured models lead to more accurate predictions, and leverage user input much more effectively than state-of-the-art independent models.

1. Introduction

In this paper we address the problem of image labeling, where the goal is to predict the relevant labels from a given annotation vocabulary for an image. This problem is also known as image classification or auto-annotation, and the label predictions can be used for clustering, (attribute-based) classification, and retrieval. Hence it is an important functionality for any multimedia content management system, stock photography database indexing, or for exploring images on photo sharing sites.

Most existing systems address the problem of image annotation either in a fully manual way (*e.g*. stock photo sites as Getty images), or in a fully automatic setting where image labels are automatically predicted without any user interaction. In the latter case most commonly used are either classifiers *e.g*. [20], ranking models *e.g*. [8], or nearest neighbor predictors [9]. Although, there are correlations in the classifier outputs, since the independent predictors use the same input images for prediction, the dependencies among the labels are generally not modeled explicitly.

In contrast to this predominant line of work, we propose structured models that take into account the dependencies among the image labels explicitly. These models are more expressive and lead to more accurate predictions of image labels. While in the setting of fully automatic image annotation the improvement is modest, the gain becomes much more significant in an interactive scenario, where a user is asked to confirm or reject some of the image labels.

Such an interactive scenario is for example useful for indexing of images for stock photography, where a high indexing quality is mandatory, yet fully manually indexing is very expensive and suffers from very low throughput. The interactive scenario offers an interesting trade-off between accuracy and manual labeling effort. In this case the label dependencies in the proposed models can be leveraged in two ways. First, the structured models are able to transfer the user input for one image label to more accurate predictions on other image labels, which is impossible with independent prediction models. Second, using structured models the system will not query, wastefully, for image labels that are either highly dependent on already provided labels, or predicted with high certainty from the image content. Through inference in the graphical model the system fuses the information from the image content and the user responses, and is able to identify labels that are highly informative once provided by the user.

We conduct experiments using three public benchmark data sets: the Scene Understanding dataset [4] (SUN'09), the dataset of the ImageCLEF'10 Photo Annotation Task [12] (ImageCLEF), and the Animals with Attributes dataset [11] (AwA). Our results without user input

SUN 09 - 5 labels	Before	Questions	After
	01 Sky		01 Rock
	02 Tree		02 Rocks
	03 Building		03 Sea
	04 Sea	Building	04 Sky
	05 Rocks	Tree	05 Sand
	06 Plant	Sea	06 Ground
	07 Ground	Rocks	07 Plant
	08 Rock	Rock	08 Person
	09 Person		09 Window
	10 Window		10 Water

AwA - 29 labels	Before	Questions	After
	01 Fast		01 Toughskin
	02 Active		02 Swims
	03 Smart		03 Arctic
	04 Meatteeth	Toughskin	04 Water
	05 Newworld	Paws	05 Fish
	06 Agility	Swims	06 Ocean
	07 Tail	Mountains	07 Fast
	08 Meat	Arctic	08 Active
	09 Strong		09 Strong
	10 Chewteeth		10 Smart

Figure 1. Interactive image annotation for images from the SUN'09 data set (left, with 5 ground truth labels), and the AwA data set (right, with 29 ground truth labels). We show the ten labels predicted with highest confidence before and after user input (green labels appear in the ground truth, red ones do not), as well as the five labels that were selected by the system to be confirmed or rejected by the user.

are comparable or better than the state-of-the-art reported on these data sets. The experiments also show that a relatively small amount of user input can substantially improve the results, in particular when we use our proposed models that capture label dependencies. To give an idea of the impact of user input, we illustrate the interactive image annotation process for two example images in Figure 1.

In addition to showing the effectiveness of structured models for interactive image labeling, we also explore how the proposed structured models can be exploited in the context of attribute-based image classification [3, 11]. The attributes are shared between different classes, and image classification proceeds by predicting the attribute values from the image, and then mapping these to class probabilities by means of a given attribute-to-class mapping. Predicting the attribute values for an image can be seen as annotating an image with a set of labels, and we use our structured models at the attribute level. The user interaction will also take place at the attribute level, but in this case the system will ask for user input on the attribute level to improve the class predictions rather than the attribute prediction. Experiments on the AwA data set show that also in this case the structured models outperform independent attribute prediction, both in automatic and interactive scenarios, and that a small amount of user input on the attributes substantially improves the classification results.

In Section 2, we discuss how our work is related to recent work on image classification and annotation. Then we present our structured prediction models in Section 3. Section 4 describes the extension to attribute-based image classification. Section 5 addresses the problem of the label elicitation in the interactive scenario. We present experimental results in Section 6, and our conclusions in Section 7.

2. Related work

Most work on image annotation, object category recognition, and image categorization has focused on methods that deal with one label or object category at a time. The function that scores images for a given label is obtained by means of various machine learning algorithms, such as binary SVM classifiers using different (non-)linear kernels [20], nearest neighbor classifiers [9], and ranking models trained for retrieval [8] or annotation [19]. Classification is more challenging when dealing with many classes, both when the aim is to assign a single label to an image from many possible ones [6], as in the case where for each image several labels should be predicted, *e.g.* all present object categories [4].

To address this difficulty, there has been a recent focus on contextual modeling. For example in object class recognition, the presence of one class may suppress (or promote) the presence of another class that is negatively (or positively) correlated, see *e.g.* [4, 7, 15]. In [15] the goal was to label the regions in a pre-segmented image with category labels, and a fully-connected conditional random field model over the regions was used. In [7] contextual modeling was used to filter the windows reported by object detectors for several categories. The contextual model includes terms for each pair of object windows that will suppress or favor spatial arrangements of the detections (*e.g.* *boat* above *water* is favored, but *cow* next to *car* is suppressed). A similar goal was pursued in [4], where a tree-structured model is used to enhance the scores of bounding boxes proposed by a discriminatively trained object detector. The presence and location of the object category in the context of all other bounding boxes from the image is modeled using the tree. The parameters of the model are learned in a generative way, from images with bounding-boxes. In our work we also use tree structured models, but over global labels using only presences and absences of the labels, and we learn the complete model discriminatively.

The interactive image annotation scenario we address in this paper is related to active learning where user input is leveraged to improve prediction models during training. In active learning for classification, the learning algorithm disposes of a number of labeled and unlabeled examples. Iteratively, a classification model is learned from the labeled ones, and then using this model the system determines which example (image) is most valuable to be labeled next by the user [16]. Such models have been used to learn from user input at different levels of granularity, *e.g.* by querying image-wide labels or precise object segmentations [18].

In our work, however, the system does not select images to be labeled at training time by a user to improve the model. Instead, for a given image at test time, it selects labels for which user-input is most valuable in order to improve predictions on the other labels of the same image.

In this paper we also apply our approach to attribute-based image classification, where an image is assigned to a given class based on a set of given attributes [3, 11]. The attributes are shared across image classes, and image classification proceeds by predicting the most likely attribute configuration that corresponds to one of the possible classes. A similar setting was recently studied in [3] for attribute-based object recognition, where each image belongs to exactly one out of many possible categories. In their work a discriminative SVM object recognition system is combined with a generative class-attribute model: for each class they independently modeled the user object attributes values reported by different users (allowing for erroneous user responses and ambiguous object-attribute relationships). To leverage user input for classification, the system asks the user to label the attribute that reduces the entropy on the class label the most (in expectation with respect to the yet unknown user response). Similarly, we also exploit user input at the level of attributes, but we learn recognition models for each attribute rather than for the object categories. This has the advantage that it allows for recognition of classes for which no training images are available, but only an attribute-based description is known, *i.e*. zero-shot classification [11]. As compared to the model of [11], we go one step further by modeling the dependencies between attribute labels. This allows us not only to improve attribute-based recognition, but also to better exploit the user input by asking more informative questions.

3. Structured image annotation models

We now describe our image annotation models, starting with tree-structured conditional models in Section 3.1, and extending to trees over groups of labels in Section 3.2.

3.1. Tree-structured model on image labels

We use a tree-structured conditional random field, and define the tree such that each node represents a label from the annotation vocabulary. Let $\boldsymbol{y} = \{y_1, \ldots, y_L\}$ denote a vector of the L label variables, which we will assume to be binary valued for sake of simplicity, *i.e*. $y_i \in \{0, 1\}$. Let $\mathcal{E} = \{e_1, \ldots, e_{L-1}\}$ define the edges in the tree over the label variables, where $e_l = (i, j)$ indicates the presence of an edge between y_i and y_j. Our basic structured model is a tree-structured conditional model on the image labels $\boldsymbol{y}$ given the image $\boldsymbol{x}$, and is defined as:

$$p(\boldsymbol{y}|\boldsymbol{x}) = \frac{1}{Z(\boldsymbol{x})} \exp -E(\boldsymbol{y}, \boldsymbol{x}), \tag{1}$$

$$E(\boldsymbol{y}, \boldsymbol{x}) = \sum_{i=1}^{L} \psi_i(y_i, \boldsymbol{x}) + \sum_{(i,j)\in\mathcal{E}} \psi_{ij}(y_i, y_j), \tag{2}$$

where

$$Z(\boldsymbol{x}) = \sum_{\boldsymbol{y}\in\{0,1\}^L} \exp -E(\boldsymbol{y}, \boldsymbol{x}), \tag{3}$$

is an image-dependent normalizing term known as the partition function, and $E(\boldsymbol{y}, \boldsymbol{x})$ is an energy function scoring the compatibility between an image $\boldsymbol{x}$ and a label vector $\boldsymbol{y}$.

To define the energy function, we use generalized linear functions for the unary potentials:

$$\psi_i(y_i = l, \boldsymbol{x}) = \boldsymbol{\phi}_i(\boldsymbol{x})^\top \boldsymbol{w}_i^l, \tag{4}$$

where $\boldsymbol{\phi}_i(\boldsymbol{x})$ is the feature function of the image and $\boldsymbol{w}_i^l$ are the weighting parameters. For the sake of efficiency we have used very compact feature functions $\boldsymbol{\phi}_i(x) = [s_i(\boldsymbol{x}), 1]^\top$, where $s_i(\boldsymbol{x})$ is an SVM score function associated with label variable y_i, however the model allows for more complex unary feature functions, *e.g*. by extending $\boldsymbol{\phi}_i(x)$ to include the results of a set of different classifiers possibly trained on different modalities or feature channels.

The pairwise potentials are defined by scalar parameters for each joint state of the corresponding labels, independent of the image input:

$$\psi_{ij}(y_i = s, y_j = t) = v_{ij}^{st}. \tag{5}$$

Since the model is tree-structured, inference is tractable and can be performed by standard belief propagation algorithms [1]. Inference is used to evaluate the partition function $Z(\boldsymbol{x})$, to find marginal distributions on individual labels $p(y_i|\boldsymbol{x})$, the pairwise marginals $p(y_i, y_j|\boldsymbol{x})$, and the most likely joint labeling state $\boldsymbol{y}^* = \arg\max_{\boldsymbol{y}} p(\boldsymbol{y}|\boldsymbol{x})$.

Finding the optimal tree structure for conditional models is generally intractable [2], therefore we resort to methods developed for generative models for finding a useful tree structure over the labels. The optimal tree structure for a generative model of a multivariate distribution can be computed using the Chow-Liu algorithm [5]. It computes the maximum spanning tree over a fully connected graph over the label variables with edge weights given by the mutual information between the label variables. We estimate the mutual information between pairs of label variables from the empirical distribution of the training data.

Having fixed a particular tree structure we learn the parameters of the unary and pair-wise potentials by the maximum likelihood criterion. Given N training images $\boldsymbol{x}_n$ and their annotations $\boldsymbol{y}_n$, we seek to maximize

$$\mathcal{L} = \sum_{n=1}^{N} \mathcal{L}_n = \sum_{n=1}^{N} \ln p(\boldsymbol{y}_n|\boldsymbol{x}_n). \tag{6}$$

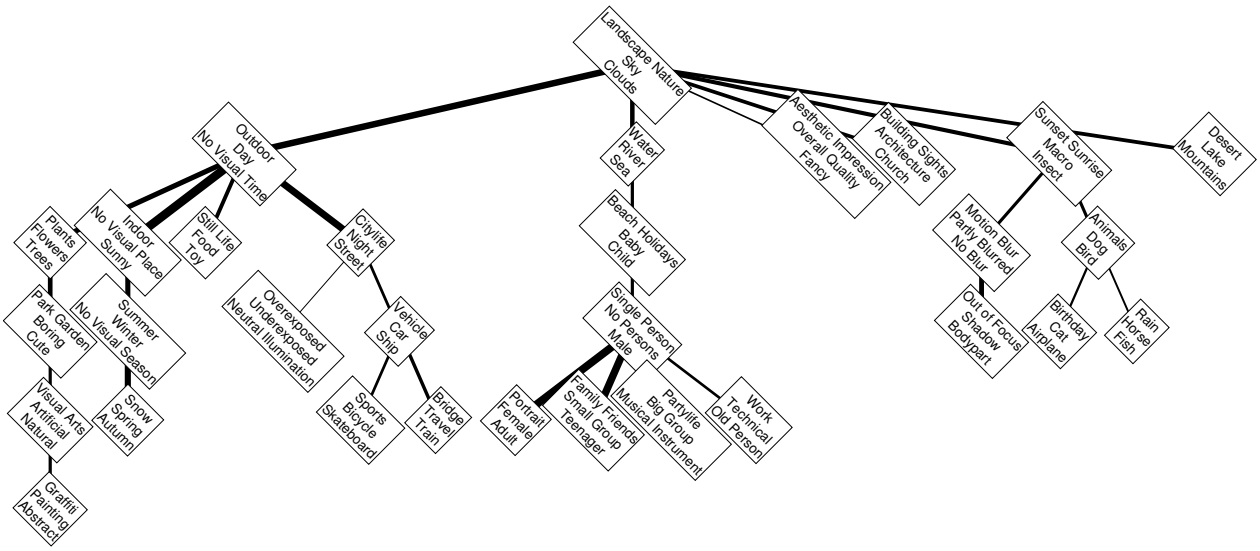

Figure 2. An example of a tree over groups of at most $k = 3$ labels on the $L = 93$ labels of the ImageCLEF data set. The edge thickness is proportional to the mutual information between the linked nodes. The root of the tree has been chosen as the vertex with highest degree.

As the energy function is linear in the parameters, the log-likelihood function is concave, and the parameters can be optimized using gradient-based methods. Computing the gradient requires evaluation of the marginal distributions on single variables, and pairs of variables connected by edges in the tree. Using y_{in} to denote the value of variable i for training image n, we have:

$$\frac{\partial \mathcal{L}_n}{\partial \boldsymbol{w}_i^l} = \Big(p(y_i = l|\boldsymbol{x}_n) - [\![y_{in} = l]\!]\Big)\boldsymbol{\phi}_i(\boldsymbol{x}_n), \quad (7)$$

$$\frac{\partial \mathcal{L}_n}{\partial v_{ij}^{st}} = p(y_i = s, y_j = t|\boldsymbol{x}_n) - [\![y_{in} = s, y_{jn} = t]\!], \quad (8)$$

where $[\![\cdot]\!]$ equals 1 if the expression is true, and 0 otherwise.

3.2. Trees over groups of label variables

To accommodate for more dependencies between labels in the model, we consider a simple extension where we group the label variables, and then define a tree over these groups. A label group can be seen as a fully connected set of variables in the graphical model; if k equals the number of labels L we have a fully connected model, in which inference is intractable. The group size k offers a trade-off between expressiveness of the model, computational tractability and the risk of overfitting on the training data.

In order to obtain the tree, we first perform agglomerative clustering based on mutual information, fixing in advance a maximum group size k, then we build the tree using the Chow-Liu algorithm as described above. In Figure 2 we show a tree with group size 3. Although not forced, semantically related concepts are often grouped together, *i.e.* *Water*, *River*, and *Sea*, or linked together in a sub-tree like the sub-tree around the *Single Person* node.

Let $\{\mathcal{G}_g\}_{g=1}^G$ denote the partition of original labels $\{1, \ldots, L\}$ into G groups, such that if $g \neq h$ then $\mathcal{G}_g \cap \mathcal{G}_h = \emptyset$, and $\bigcup_{g=1}^G \mathcal{G}_g = \{1, \ldots, L\}$. With each group of variables, we associate a new variable $\boldsymbol{y}_g$ that takes as values the product space of the values of the labels in the groups. Thus, for groups of k binary labels, $\boldsymbol{y}_g$ takes 2^k values, and there is a one-to-one mapping between the values of the variables in the group and the value of the group variable.

The unary potentials are defined as in Eq. (4), where y_i is replaced with y_g . Similarly, $\boldsymbol{\phi}_g(\boldsymbol{x}) = [\{s_i(\boldsymbol{x})\}_{i \in \mathcal{G}_g}, 1]$ becomes the extended vector of SVM scores associated with the image labels in the group. The pairwise potential of Eq. (5), now links groups of k binary variables, and hence will be defined by 2^{2k} scalars. Therefore the cost of message passing algorithms scales with $O(G2^{2k})$. In order to maintain tractable inference the group sizes should be fairly small, in our experiments we use $k \leq 4$.

In addition we consider a mixture of trees with different group sizes k. We train the models independently, and then average the predictions of the individual models. Alternatively, the mixing weights can be learned concurrently while learning the trees, possibly improving results.

While we only use trees over (groups of) labels, the proposed framework can easily be extended to other graph structures, provided that the tree-width of the graphical model is relatively low to ensure tractable inference. Similarly, the binary label case we considered here can be trivially extended to image labels taking one among three or more mutually exclusive values. In principle we could also train our models using max-margin methods [17], but in that framework it is less clear how to define label elicitation strategies, such as the ones presented in Section 5.

4. Attribute-based image classification

Attribute-based image classification [3, 11] refers to a classification paradigm where an image is assigned to a given class $z \in \{1, \ldots, C\}$ based on a set of attribute values. An image belongs to exactly one class, but attributes are shared between different classes. For example, in the AwA data set different animals are defined in terms of attributes such as *has stripes, has paws, swims*, etc. The advantages of such a system is that it can recognize unseen classes based on an attribute-level description, and that the attribute representation can in principle encode an exponential number of classes. By sharing the attributes between different classes, classifiers for each of the attributes can be learned by pooling examples of different classes which increases the number of training examples per classifier as compared to the number of examples available for the individual classes.

Here, we apply our structured prediction model at the level of attributes, *i.e*. we learn a tree structured model over attributes, and the binary values y_i now refer to the presence or absence of an attribute for an image. As in [11], we assume a deterministic mapping between attributes and the C object classes is given, and denote the attribute configuration of class c by $\boldsymbol{y}_c$. We define the distribution over image classes by the normalized likelihoods of the corresponding attribute configurations

$$p(z=c|\boldsymbol{x}) = \frac{p(\boldsymbol{y}_c|\boldsymbol{x})}{\sum_{c'=1}^{C} p(\boldsymbol{y}_{c'}|\boldsymbol{x})} = \frac{\exp -E(\boldsymbol{y}_c, \boldsymbol{x})}{\sum_{c'=1}^{C} \exp -E(\boldsymbol{y}_{c'}, \boldsymbol{x})}. \quad (9)$$

Note that the evaluation of $p(z|\boldsymbol{x})$ does not require belief-propagation, it suffices to evaluate $E(\boldsymbol{y}_c, \boldsymbol{x})$ for the C attribute configurations $\boldsymbol{y}_c$, since the partition function $Z(\boldsymbol{x})$ of Eq. (3) cancels from both numerator and denominator.

When using our model as such, we observe that some classes tend to be much more often predicted than others, and the prediction errors are mainly caused by assigning images to these over-predicted classes. As this also holds for the independent attribute prediction model, we assume the reason might be that some classes have rare (combinations of) attribute values. In order to overcome this we introduce a correction term, u_c, for each class to ensure that all classes will be predicted equally often in expectation. We redefine therefore the class prediction model of Eq. (9) as

$$p(z = c|\boldsymbol{x}) \propto \exp\big(- E(\boldsymbol{y}_c, \boldsymbol{x}) - u_c\big), \quad (10)$$

and set the u_c such that on the training data we have $\sum_n p(z = c|\boldsymbol{x}_n) = n_c$ for all classes, with n_c the number of images in class c. To find the values of u_c we use a procedure similar to logistic regression training. In the case of zero-shot learning the test classes have not been seen among the training images, therefore we do not have the class counts n_c available, and we set $n_c = N/C$.

5. Label elicitation strategies

In this section we describe our interactive image annotation scenario, where a user is asked iteratively to reveal the value of selected labels. While a random choice of labels is possible, and the system can take advantage of those values, we propose a label election strategy whose aim is to minimize the uncertainty of the remaining labels or the class label given the test image.

5.1. Label elicitation of image annotation

Our goal is to select the label y_i for which knowing its ground truth value minimizes the uncertainty on the other labels. To achieve this, we propose to minimize the entropy of the distribution on the label vector $\boldsymbol{y}$ given the user input for one label y_i, by varying i which indicates which label will be set by the user.

Let us use y_i^l to denote $y_i = l$, and $\boldsymbol{y}_{\setminus i}$ to denote all label variables except y_i. Then, given y_i^l the uncertainty on other labels, $\boldsymbol{y}_{\setminus i}$ is quantified by the entropy

$$H(\boldsymbol{y}_{\setminus i}|y_i^l, \boldsymbol{x}) = -\sum_{\boldsymbol{y}_{\setminus i}} p(\boldsymbol{y}_{\setminus i}|y_i^l, \boldsymbol{x}) \ln p(\boldsymbol{y}_{\setminus i}|y_i^l, \boldsymbol{x}). \quad (11)$$

However, the value of y_i is not known prior to the moment that it is set by the user. Therefore, we evaluate the expectation of Eq. (11), *i.e*. we want the user to set variable y_i that minimizes the expected conditional entropy

$$H(\boldsymbol{y}_{\setminus i}|y_i, \boldsymbol{x}) = \sum_l p(y_i = l|\boldsymbol{x}) H(\boldsymbol{y}_{\setminus i}|y_i = l, \boldsymbol{x}). \quad (12)$$

Given the basic identity of conditional entropy, see *e.g*. [1],

$$H(\boldsymbol{y}|\boldsymbol{x}) = H(y_i|\boldsymbol{x}) + H(\boldsymbol{y}_{\setminus i}|y_i, \boldsymbol{x}), \quad (13)$$

and as $H(\boldsymbol{y}|\boldsymbol{x})$ does not depend on the selected variable y_i, we can deduce that minimizing Eq. (12) for y_i is equivalent to maximizing $H(y_i|\boldsymbol{x})$ over i. Hence, we select the label variable y_i that has maximum marginal entropy $H(y_i|\boldsymbol{x})$.

In order to select a collection of labels to be set by the user we proceed sequentially by first asking the user to set only one label. We then repeat the procedure while conditioning on the labels already provided by the user. Note that selecting a group of labels at once is another possibility, nevertheless suboptimal as it cannot leverage information contained in the user input in the selection procedure.

5.2. Attribute elicitation for image classification

In the case of attribute-based image classification we could use the same strategy as above at attribute level. However, since the final aim is to improve the class prediction we use an attribute elicitation criterion that is geared towards minimizing uncertainty on the class label, rather than uncertainty at the attribute level. The main insight is that

the information obtained from a revealed attribute value depends on the agreement among the classes on this attribute. If some of the probable classes do not agree with the observed value it will rule out the classes with a contradicting attribute value and concentrate the probability mass on the compatible classes. Therefore, any informative question will at least rule out one of the possible classes, and thus at most $C-1$ attributes need to be set by the user.

In order to see which attribute should be set by the user we minimize the conditional class entropy $H(z|y_i,\boldsymbol{x})$. Using the identity

$$H(z,\boldsymbol{y}|\boldsymbol{x}) = H(y_i|\boldsymbol{x}) + H(z|y_i,\boldsymbol{x}) + H(\boldsymbol{y}_{\setminus i}|z,y_i,\boldsymbol{x}), \quad (14)$$

we make the following observations: (i) The left-hand-side of the equation does not depend on the choice of attribute y_i to elicit. (ii) The last term $H(\boldsymbol{y}_{\setminus i}|z,y_i,\boldsymbol{x})$ equals zero, since for each class there is a unique setting of the attribute values. Therefore, selecting the attribute to minimize the remaining entropy on the class label is equivalent to selecting the attribute with the largest marginal entropy $H(y_i|\boldsymbol{x})$.

Note that in the attribute-based classification model $p(y_i|\boldsymbol{x})$ differs from the image annotation model. Here $p(y_i|\boldsymbol{x})$ is implicitly defined through Eq. (10) which essentially rules-out all attribute configurations, except the ones that correspond to one of the C classes. Therefore, we have

$$p(\boldsymbol{y}|\boldsymbol{x}) = \sum_c p(z=c|\boldsymbol{x})[\![\boldsymbol{y}=\boldsymbol{y}_c]\!], \quad (15)$$

$$p(y_i|\boldsymbol{x}) = \sum_{\boldsymbol{y}_{\setminus i}} p(\boldsymbol{y}|\boldsymbol{x}) = \sum_c p(z=c|\boldsymbol{x})[\![y_i=y_{ic}]\!]. \quad (16)$$

where y_{ic} denotes the value of attribute i for class c. In particular, for binary attributes we have

$$p(y_i=1|\boldsymbol{x}) = \sum_c p(z=c|\boldsymbol{x}) y_{ic}, \quad (17)$$

As above, sequences of user queries are generated progressively by conditioning on the image and all the attribute labels given so far to determine the next attribute to query.

6. Experimental evaluation

Below, we first present our experimental setup, and then the results for automatic and interactive image annotation, followed by results on attribute-based image classification.

6.1. Data sets and implementation details

In our experiments we use three recent public data sets, see Table 1 for some basic statistics of the data sets.

The *ImageCLEF'10* data set is a subset of the MIR-Flickr data set [10] used in the ImageCLEF Photo Annotation task in 2010 [12] as training set. The images are labeled with 93 concepts, see Figure 2. As well as the images, the Flickr-tags belonging to each image are given. We use the same features as in our system that won the Photo Annotation task: a concatenation of the improved Fisher vector representation [13] computed over SIFT and color features, and a binary vector denoting the presence of the most common Flickr-tags. We split the data into five folds and report results averaged over the folds. For the sake of clarity we omit standard deviations since they are small compared to the differences between prediction methods.

Table 1. Basic statistics of the three data sets.

	ImageCLEF	SUN'09	Animals w.A.
# Train images	6400	4367	24295
# Test images	1600	4317	6180
# Labels	93	107	85
Train img/label	833	219	8812
Train label/img	12.1	5.34	30.8

The *SUN'09* data set was introduced in [4] to study the influence of contextual information on localization and classification, we compare to their classification results. We use the same image features as for the ImageCLEF data set. For both data sets we use linear SVM classifiers, with $C=1$.

The *Animals with Attributes* (AwA) [11] contains images of 50 animal classes, and a definition of each class in terms of 85 attributes. We follow [11], using the provided features, the same sum of RBF-χ^2 kernels, regulation parameter $C=10$, and the same 40 train and 10 test classes.

In the independent prediction models we use confidence values obtained by a sigmoid function over the SVM scores. This allows us to select labels for user input and to rank labels by confidence values for a given image. To learn our tree-structured models, or sigmoids for the independent models, we use a method similar to Platt scaling [14]: the train set is split into five folds. For each fold, f, we obtain the classification scores by training SVMs on the remaining folds. For test images we use SVM scores obtained by training on all training images.

6.2. Fully automatic image annotation

We compare the influence of the structured models in the setting of fully automatic prediction. As evaluation measures we use average precision (AP), which indicates the retrieval performance of a label over the dataset. We then consider the mean of AP over all labels (MAP). We also use AP of labels at image level (iAP), which is the average precision of the correct labels for each image. This is a performance of annotating a single image, and therefore we average it over all images to obtain iMAP. This performance measure correlates to the amount of corrections needed to obtain a completely correct image labeling.

In the top row of Figure 3 we show the performance of each data set for fully automatic prediction. We compare

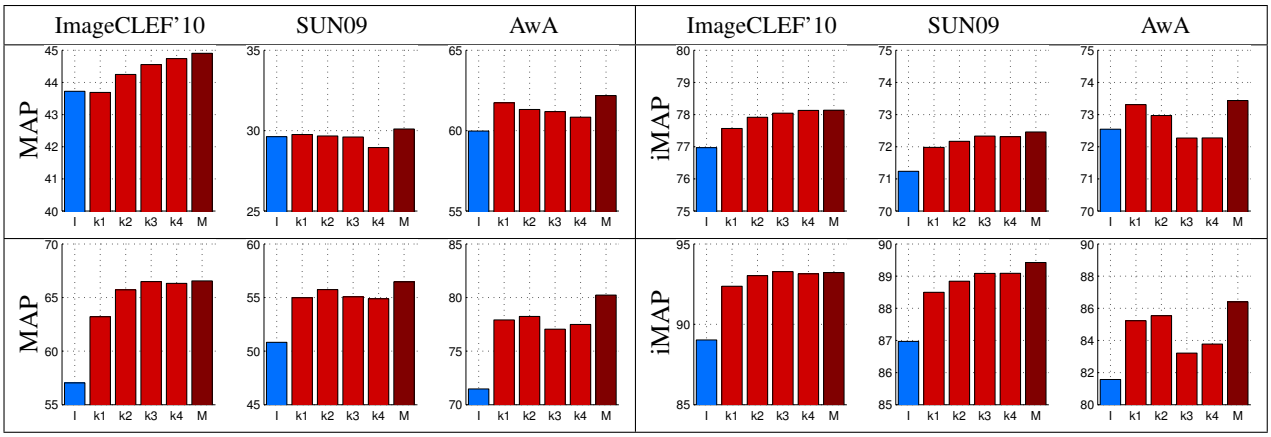

Figure 3. An overview of the performance of the different models in MAP and iMAP on the three data sets. In the first row the performance of the fully automated prediction setting is shown, while the second row shows the performance of an interactive setting with 10 questions. We compare results of the independent model (yellow), the trees with group sizes $1 \leq k \leq 4$ (light-red), and the mixture of trees (dark-red).

the independent prediction model against trees with group sizes $1 \leq k \leq 4$, and to the mixture of the tree models. To the best of our knowledge our independent classifiers (the yellow bars in Figure 3) have (near) state-of-the-art performance. For the SUN09 and AwA data sets we are the first to report accuracy in MAP over image labels/attributes.

From the results we see that most structured prediction models slightly outperform the independent model for both MAP and iMAP. The performance differences between the models with different group size k, should be seen as trade off between model expressiveness and overfitting on the training data. The mixture model generally performs best, or is comparable to the best performing model.

The improvement of the structured models is rather small, but note that the trees only propagate visual information, which is already captured very well by the independent SVM classifiers. In the next section, where we consider interactive image annotation, the trees are more useful since in that case they also propagate the user input.

6.3. Interactive image annotation

In this setting we simulated the user input by assigning the ground truth value to labels iteratively selected for user input. In the bottom row of Figure 3 we show the performance of the different systems after setting ten image labels. As expected, the structured models benefit more from the user input, since they can propagate the information to update their belief of other labels. Also in this setting the mixture of trees performs best, or is comparable to the best model. The improvements of the tree structured models over the independent model are much larger in this case. The user input makes some of the label variables observed, these variables now no longer propagate visual information, but they send messages based on their observed value to the variables connected to them. This new information translates to better predictions on the unknown labels in the tree.

Table 2. Zero-shot attribute-based classification accuracy of the independent and mixture of trees models. Initial results, and after user input for one up to eight selected attributes.

	Init	1	2	3	4	5	6	7	8
Indep	36.5	53.1	68.5	77.8	85.1	90.6	94.5	97.7	99.4
Mixt	38.7	55.3	72.3	84.8	92.4	96.9	99.0	99.8	100.0

In Figure 4 (left and middle) we further show the performance on the ImageCLEF database for the independent predictors and our mixture model, from no user input to complete user-input on all labels. We can see that our method achieves perfect labeling after significant fewer steps than the independent predictors. In order to illustrate the benefit of the proposed entropy-based criteria, we also show the labeling performance when randomly selecting labels for user input. Observe that both the structured model, and the label elicitation mechanism help to improve performance.

In Figure 4 we also show the performance of our models compared to the hierarchical context method of [4], using the evaluation method proposed therein. The results show that even our baseline method clearly outperforms their hierarchical context model (HContext) which also relies on object bounding boxes during training. This is partially explained by the stronger improved Fisher vectors features we use instead of GIST. The differences between the independent and structural methods become larger for more difficult images (larger N, see figure), and after more user input.

6.4. Attribute-based prediction of unseen classes

We experimented with the AwA data set to evaluate the performance of our models in predicting class labels of images from unseen classes based on the class specific configuration of the 85 attributes. To compare our approach to

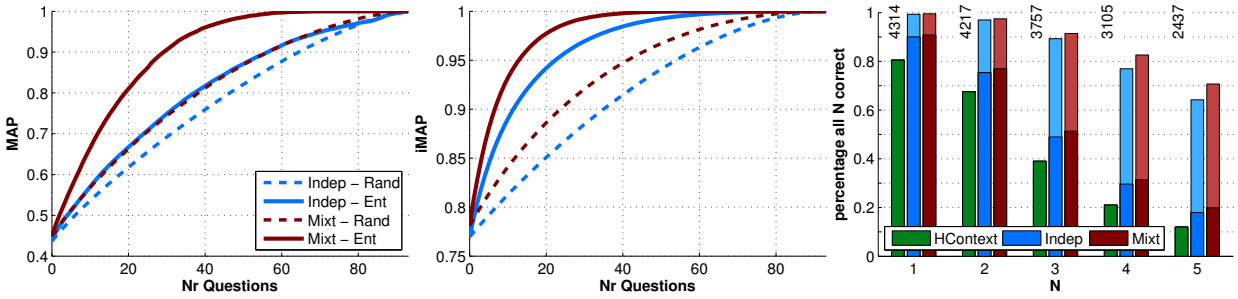

Figure 4. Left and middle: MAP and iMAP scores as function of the number labels set by the user on ImageCLEF'10. Right: percentage of images with at least N labels (the number of such images in top) for which the top N predicted labels are all correct on the SUN'09 data set. The dark bars show the performance for automatic prediction, the light bars on top show the performance after user input for 10 labels.

the state-of-the art, we use the same setting and the same measure (mean of the diagonal of the normalized confusion matrix) as in [11]. Table 2 shows the performance of the independent model and the mixture model, after asking up to eight questions. Note that the tree structured models learn attribute dependencies for the train classes which are different from the test classes, *i.e*. during testing combinations of attributes are seen which have never been seen before. Still, these models significantly improve over the results of the independent model. This is also reflected in the average number of attributes set by the user before the correct class is ranked first: 1.82 ± 2.06 for the independent model, and 1.54 ± 1.67 for the mixture of trees model.

7. Conclusion

We have introduced a class of structured models for image labeling and have shown that it can be successfully applied to different application scenarios such as automatic and semi-automatic image annotation and attribute-based image classification. While these models offer moderate improvements over independent baseline models, their real power is exploited particulary in the interactive setting. In this case, where the system asks a user to set the value of a small number of labels, the proposed models are able to transfer the user input to more accurate predictions on the other image labels. A similar trend of stronger improvement with more user input is also observed in the case of attribute-based image classification.

References

[1] C. Bishop. *Pattern recognition and machine learning*. Spinger-Verlag, 2006.

[2] J. Bradley and C. Guestrin. Learning tree conditional random fields. In *ICML*, 2010.

[3] S. Branson, C. Wah, F. Schroff, B. Babenko, P. Welinder, P. Perona, and S. Belongie. Visual recognition with humans in the loop. In *ECCV*, 2010.

[4] M. Choi, J. Lim, A. Torralba, and A. Willsky. Exploiting hierarchical context on a large database of object categories. In *CVPR*, 2010.

[5] C. Chow and C. Liu. Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, 14(3):462–467, 1968.

[6] J. Deng, A. Berg, K. Li, and F.-F. Li. What does classifying more than 10,000 image categories tell us? In *ECCV*, 2010.

[7] C. Desai, D. Ramanan, and C. Fowlkes. Discriminative models for multi-class object layout. In *ICCV*, 2009.

[8] D. Grangier and S. Bengio. A discriminative kernel-based model to rank images from text queries. *PAMI*, 30(8):1371–1384, 2008.

[9] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid. Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation. In *ICCV*, 2009.

[10] M. Huiskes and M. Lew. The MIR Flickr retrieval evaluation. In *ACM MIR*, 2008.

[11] C. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*, 2009.

[12] S. Nowak and M. Huiskes. New strategies for image annotation: Overview of the photo annotation task at ImageCLEF 2010. In *Working Notes of CLEF*, 2010.

[13] F. Perronnin, J. Sánchez, and T. Mensink. Improving the Fisher kernel for large-scale image classification. In *ECCV*, 2010.

[14] J. Platt. Probabilities for SV machines. In *Advances in Large Margin Classifiers*, 2000.

[15] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie. Objects in context. In *ICCV*, 2007.

[16] B. Settles. Active learning literature survey. Technical Report 1648, University of Wisconsin-Madison, 2009.

[17] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *JMLR*, 6:1453–1484, 2005.

[18] S. Vijayanarasimhan and K. Grauman. Multi-level active prediction of useful image annotations for recognition. In *NIPS*, 2009.

[19] J. Weston, S. Bengio, and N. Usunier. Large scale image annotation: Learning to rank with joint word-image embeddings. In *ECML*, 2010.

[20] J. Zhang, M. Marszałek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *IJCV*, 73(2):213–238, 2007.