Image and Audio Annotation: Approximate Inference in Dense Conditional Random Fields

Andrew C. Miller
Department of Computer Science
Brown University
amiller@cs.brown.edu

Erik B. Sudderth (Advisor)
Department of Computer Science
Brown University
sudderth@cs.brown.edu

Abstract

In the task of image and audio annotation, it is typical for labels to be assigned independently. While some correlations may be modeled, it is rare for all interdependencies between labels to be considered. This is because exact inference on a dense, cyclical graphical model is often intractable. This paper applies an approximate inference method, log-determinant relaxation (LDR), to a fully connected conditional random field (CRF) to perform automatic image and audio annotation. The proposed model estimates the conditional distribution of all labels in a vocabulary given the audio or image features. Two other discriminative models incorporating a varying degree of context are used as a baseline for comparison. We show that introducing context to CRFs improves annotation performance, and can be made tractable with LDR. At the time of writing, the application of LDR to a discriminative model is novel.

1 Introduction

Text, images, and music are three common mediums of information through which people communicate their style, emotion, origins, and other distinctly human ideas. It is often the case that in discussing an image, document or song, one reduces the object to a set of words. Often these words will be ambiguous or even subjective, leading to disagreements on what is 'correct'. However, there are many informative and often agreed upon labels that one could apply to these objects. Words that describe a piece of music could reference its instrumentation, style, timbre, emotion, and its typical use; one could convey information about an image by listing objects that appear in it; documents are often reduced to a list of relevant topics. This descriptive agreement suggests that a pattern recognition system can learn the relationship between these signals and their descriptions (represented by a list of annotations) [16].

In the music information retrieval community, audio annotation is seen as only part of the solution to a larger problem. The main goal is to create systems that efficiently store and retrieve songs from large databases of musical content [17]. With that goal in mind, a few different query systems have been proposed. Collaborative filtering approaches have been implemented using a song's metadata (artist, reviews, ratings - any non acoustic representation), however these systems will either ignore the musical content, or rely on manual annotation [16]. Research has also been conducted on 'query by similarity' databases. These take audio based queries and return the most 'similar' songs in the database [19]. Systems have been created to take queries in the form of humming, tapping,

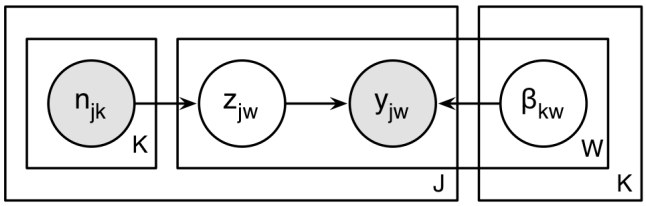


Figure 1: Directed graphical model representation of Codeword Bernoulli Average (CBA). Shaded nodes represent observed variables, unshaded nodes represent hidden variables. J is the number of songs, K is the number of codewords/features, and W is the number of unique tags. Each tag y_{jw} depends only on a latent variable z_{jw} (which depends only on the features of song j, $\mathbf{n}_j$) and independent parameters β_{kw} learned through EM. Tags are generated independently, and the generative process does not model any semantic context.

beatboxing, and even melodic contours. However, it is difficult for an untrained listener to recreate musical characteristics of a song. Because of this limitation, 'query by text' systems have been proposed - systems that learn a relationship between acoustic features and words from a dataset of annotated audio tracks [17]. Similar reasons motivate 'query by text' systems for documents and images.

Describing data with a list of meaningful annotations is a necessary step in categorizing and retrieving relevant information. This is one of the reasons why the task of annotating data is a well studied problem in computer vision, audition, and document classification [4, 11, 16].

Many audio tagging systems treat the annotation task as a set of binary classification problems, using standard classifiers such as the Support Vector Machine [9, 15], or AdaBoost [2] to make decisions and provide confidence scores. Other approaches have used generative models to capture the annotation procedure, such as a Gaussian Mixture Model (GMM) [17, 19], Codeword Bernoulli Average (CBA) [7], or simply naive Bayes [16]. In all of these approaches, each tag is classified or assumed to be generated independently with respect to every other tag. Figure 1 shows the graphical model representation of CBA, and highlights its independent tag generation.

Introducing semantic context is one way to enrich these annotation models. Semantic context (or just context) requires access to the referential meaning of the object [3]; that is, it considers the presence or absence of other annotations. Independent classifiers use only feature data to make a decision, which effectively ignores context. For example, independent classifiers do not take into account that tables tend to appear in the same images as chairs, or that boats are rarely pictured with trains. One way to incorporate semantic context in data annotation is to model dependencies between annotations. This is equivalent to adding edges between label variables in a graphical model, which can cause the model to grow exponentially in size and complexity. This can quickly render parameter estimation and exact inference intractable.

To avoid this cost, some image systems have used Monte Carlo methods, such as importance sampling, to run inference on their densely correlated annotation models [11]. The performance of these stochastic methods is often slow and difficult to assess. Importance sampling relies on a heuristic proposal distribution, while some MCMC methods may be prohibitively slow to converge. This paper explores another class of approximate inference algorithms, known as variational methods.

In this paper, we compare three discriminative models incorporating a varying degree of semantic context. The three models can be described graphically by conditional random fields (CRFs) with differing structure [14]. The first model is a baseline for comparison, an edgeless CRF that captures no context. In the second model, we restrict the structure of the CRF to be a tree, which incorporates some, but not all, dependence between labels. The third model is a fully connected CRF, for which exact inference and parameter estimation is intractable. We use an approximate inference method that has shown promise in densely correlated Markov random fields.

The remainder of this paper is organized as follows: Section 2 provides the problem setup, a brief description of undirected graphical models, conditional random fields (CRFs), and the notation used

in the rest of the paper. Section 3 describes learning and inference in the three different models. Experimental results on two real datasets from the computer audition community and the computer vision community are discussed in section 4.

2 CRFs and Annotation

This section begins by properly formulating the annotation problem. For the sake of clarity, all models will be discussed in the context of audio annotation (using songs and tags).

Our data is defined as follows: we are given a corpus of N songs and a vocabulary of V tags. Each song i will be represented by a vector of M features, notated as $\mathbf{x}_i \in \mathbb{R}^M$. The tags associated with each song will be encoded by a vector of length V, denoted by $\mathbf{y}_i \in \{-1, 1\}^V$, where y_{iv} indicates if some tag v applies to song i. Given the features of a new song, $\mathbf{x}_j$, the goal is to compute the marginal likelihood that some tag v applies to song j, $p(v \mid x_j)$, for all V tags.

2.1 Discriminative Models

All of the techniques presented in this paper fall into the category of discriminative modeling. That is, the conditional distribution $p(\mathbf{y} \mid \mathbf{x})$ is directly modeled. Furthermore, these models can be described by a pairwise conditional random field (CRF). A CRF is defined by an undirected graph $G = (V, E)$ where $V = \{1, ..., V\}$ is the vertex set, and E is the edge set. The vertex set denotes the variables of interest (in this case $\mathbf{Y}$, our tags), where each undirected edge $(s, t) \in E$ denotes a dependence between variables Y_s and Y_t. In a pairwise CRF, the conditional distribution can be factored

$$p(\mathbf{y} \mid \mathbf{x}) \propto \prod_{s \in V} \psi_s(y_s, \mathbf{x}) \prod_{(s,t) \in E)} \psi_{st}(y_s, y_t, \mathbf{x}) \tag{1}$$

where ψ_s and ψ_{st} are node and edge factors, or potentials. Although they have no concrete probabilistic interpretation, the output of these functions is restricted to the positive real numbers. In order to reduce the parameter space, we restrict the form of the potential functions such that $\psi_s(y_s, \mathbf{x}) = e^{[\theta_s^{\mathrm{T}} \mathbf{x}] y_s}$ and $\psi_{st}(y_s, y_t, \mathbf{x}) = e^{[\theta_{st}^{\mathrm{T}} \mathbf{x}] y_s y_t}$. This allows us to rewrite the conditional distribution as an Ising model

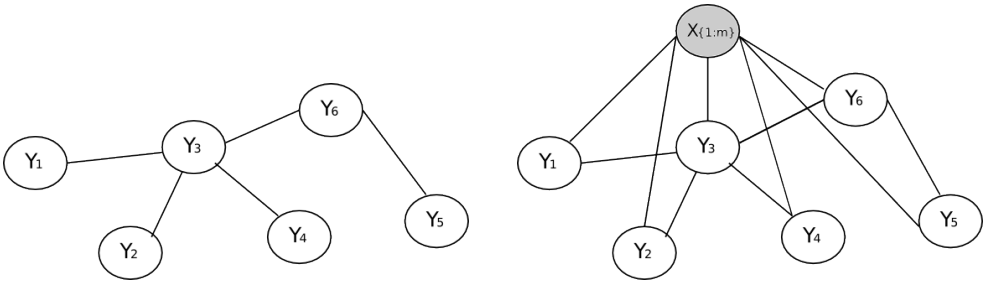
$$p(\mathbf{y} \mid \mathbf{x}) = \exp\left\{ \sum_{s \in V} [\theta_s^{\mathrm{T}} \phi_s(\mathbf{x})] y_s + \sum_{(s,t) \in E} [\theta_{st}^{\mathrm{T}} \phi_{st}(\mathbf{x})] y_s y_t - A(\theta, \mathbf{x}) \right\} \tag{2}$$

where the quantity

$$A(\theta, \mathbf{x}) = \log \sum_{y \in Y^n} \exp\left\{ \sum_{s \in V} [\theta_s^{\mathrm{T}} \phi_s(\mathbf{x})] y_s + \sum_{(s,t) \in E} [\theta_{st}^{\mathrm{T}} \phi_{st}(\mathbf{x})] y_s y_t \right\} \tag{3}$$

is the log partition function, which ensures that the conditional distribution is properly normalized, and the function $\phi_s(\mathbf{x})$ maps the observed data to a vector of M features. In this CRF formulation, features are shared between each node and pair of nodes, such that the parameters $\theta_s \in \mathbb{R}^M$ and $\theta_{st} \in \mathbb{R}^M$ are one to one with the vector features. However, it is not necessary for $\mathbf{x}$ in the first node term to be equal to $\mathbf{x}$ in the second edge term. In section IV, models with only a single bias edge feature are compared, that is $\phi_{st}(\mathbf{x}) = 1$.

Figure 2a shows a tree-structured CRF, directly modeling the conditional distribution $p(y \mid x)$, whereas figure 2b shows the corresponding Markov random field (MRF) with shared features. Although directly modeling the conditional distribution only permits conditional inference (which is sufficient for the task of classification), it allows for an implicitly rich model of the joint distribution that would otherwise be intractable to estimate.



(a) CRF tree for $y = (y_1, ..., y_6)$ tags. This is the structure of variables y when conditioned upon some observed data x. Independent models such as logistic regression or CBA would have no edges.

(b) MRF modelling the joint distribution $p(\mathbf{x}, \mathbf{y})$. Notice that the CRF vastly reduces the model complexity.

Figure 2: CRF and factor graph.

Applying this factored conditional distribution to an independently drawn dataset, D, with N examples, the log likelihood is given by

$$l(D) = \sum_{i=1}^{N} \left\{ \sum_{s \in V} [\theta_s^{\mathrm{T}} \phi_s(\mathbf{x}^{(i)})] y_s^{(i)} + \sum_{(s,t) \in E} [\theta_{st}^{\mathrm{T}} \phi_{st}(\mathbf{x}^{(i)})] y_s^{(i)} y_t^{(i)} - A(\theta, \mathbf{x}^{(i)}) \right\}. \tag{4}$$

The maximum likelihood parameter estimate, θ_{MLE}, can be found by differentiating the log likelihood and setting it equal to zero. Differentiating with respect to some weight θ_s gives

$$\frac{\partial l(D)}{\partial \theta_s} = \sum_{i=1}^{N} \left\{ \phi_s(\mathbf{x}^{(i)}) y_s^{(i)} - \frac{\partial A(\theta, \mathbf{x})}{\partial \theta_s} \right\} \tag{5}$$

where the partial derivative of the log partition function reduces to

$$\begin{aligned} \frac{\partial A(\theta, \mathbf{x})}{\partial \theta_s} &= \sum_{\mathbf{y} \in Y^V} p(\mathbf{y} \mid \mathbf{x}) \; \phi_s(\mathbf{x}^{(i)}) y_s \\ &= \mathbb{E}_\theta[\phi_s(\mathbf{x}^{(i)}) y_s] \end{aligned} \tag{6}$$

giving

$$\frac{\partial l(D)}{\partial \theta_s} = \sum_{i=1}^{N} \left\{ \phi_s(\mathbf{x}^{(i)}) y_s^{(i)} - \mathbb{E}_\theta[\phi_s(\mathbf{x}^{(i)}) y_s] \right\}. \tag{7}$$

Setting this to zero yields the equality

$$\begin{aligned} \frac{1}{N} \sum_{i=1}^{N} \phi_s(\mathbf{x}^{(i)}) y_s^{(i)} &= \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_\theta[\phi_s(\mathbf{x}^{(i)}) y_s] \\ \mathbb{E}_D[\phi_s(\mathbf{x}) y_s] &= \mathbb{E}_{\theta|D}[\phi_s(\mathbf{x}) y_s] \end{aligned} \tag{8}$$

which intuitively states that the expected value of the features under the data should equal the expected value of the features under the model. Because the log-likelihood of a log-linear CRF is differentiable and jointly convex in the node and edge parameters, we can use numerical methods to learn θ_{MLE}.

3 Model Comparison and Training

3.1 Indpendent CRF

The independent CRF assumes no correlations between nodes, thus the graph has no edges. This model can be equivalently described as V independent logistic regression models, one for each tag.

The likelihood of each tag given some song features $\mathbf{x}$ can be written as

$$p(y_s \mid \mathbf{x}) = \frac{\exp\left\{[\theta_s^{\mathrm{T}}\mathbf{x}]y_s\right\}}{\sum_{y \in Y_s} \exp\left\{[\theta_s^{\mathrm{T}}\mathbf{x}]y\right\}} \tag{9}$$

which, because our model parameters are binary, can be simplified to

$$p(y_s \mid \mathbf{x}) = \sigma(2[\theta_s^{\mathrm{T}}\mathbf{x}]y_s) := \frac{1}{1 + \exp(-2[\theta_s^{\mathrm{T}}\mathbf{x}]y_s)}. \tag{10}$$

This is the logistic function, a commonly used 'squashing' function that maps $\mathbb{R} \to [0, 1]$. Model estimation and inference can run independently for each tag.

3.2 Tree-structured CRF

The motivation for a tree-structured CRF model is that it can capture some of the correlations between tags while exact inference with belief propagation remains tractable. To determine the model structure, we do the following:

1. Start with a fully connected graph.
2. Weight each edge of the graph with the empirical mutual information between each tag, defined as
$$\hat{I}(Y_s; Y_t) = \sum_{y_s \in Y_s} \sum_{y_t \in Y_t} \hat{p}(y_s, y_t) \log\left(\frac{\hat{p}(y_s, y_t)}{\hat{p}(y_s)\hat{p}(y_t)}\right)$$
where
$$\hat{p}(y_s) = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}(y_s^{(i)}, y_s) \qquad \text{and} \qquad \hat{p}(y_s, y_t) = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}(y_s^{(i)}, y_s)\mathbf{1}(y_t^{(i)}, y_t)$$
where value N is the number of examples in the dataset, and $\mathbf{1}(a, b)$ returns 1 if $a = b$ and 0 otherwise.
3. Compute the maximum weight spanning tree of this weighted graph. Use this tree as the model structure for the conditional distribution.

While this tree optimizes the likelihood of the data, it is not necessarily the best discriminative tree structure for the model; there is no simple closed-form way to find that [5].

Figure 3 visualizes the maximum weight spanning tree for the 20 image tags used in the PASCAL VOC2009 dataset. Once this structure is determined, the conditional distribution is described by equation (2).

Inference in a tree structured CRF can be performed using belief propagation, also called the sum-product algorithm. Belief propagation is a dynamic programming algorithm that efficiently computes the exact marginal distribution for each node in a cycle-free graph. Each node receives messages from its neighbors, where these messages convey information about the node's marginal distribution. In our discriminative Ising model, the messages take the form

$$M_{t \to s}(y_s) = \sum_{y_t \in Y_t} \psi_{st}(y_s, y_t, \mathbf{x})\psi_t(y_t, \mathbf{x}) \prod_{u \in \Gamma(t) \setminus s} M_{u \to t}(y_t) \tag{11}$$

where $\Gamma(t)$ indicates the set of neighbors of node t, and $\Gamma(t)\setminus s$ indicates all neighbors except s. The local belief at node s is obtained by incorporating all of the information passed to node s from each of its neighbors. It takes the form

$$b(y_s) \propto \psi_s(y_s, \mathbf{x}) \prod_{u \in \Gamma(s)} M_{u \to s}(y_s). \tag{12}$$

In our cycle-free model, $b(y_s)$ is equivalent to $p(y_s \mid \mathbf{x})$, the likelihood used for classification. In a tree structured CRF, messages will work their way from the leaf nodes, to the root, and then back down. A loopy version of belief propagation can be applied to graphs with cycles, however its exact properties are lost. Another variational technique is applied to cyclical graphs in this paper.

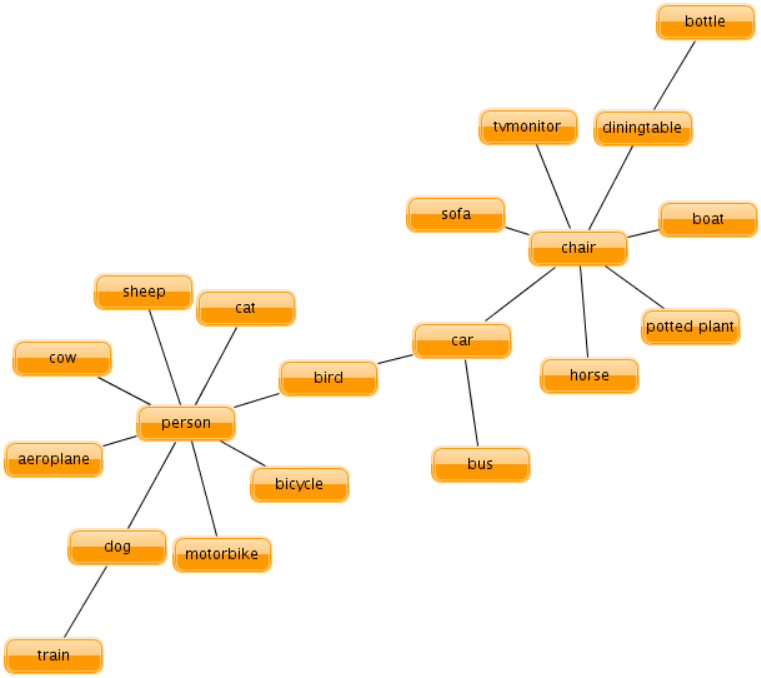


Figure 3: Max-weight-spanning tree for the pascal VOC2009 image dataset.

3.3 Dense CRFs

The fully connected pairwise CRF directly models correlations between all pairs of tags. However, due to the extremely dense and cyclical structure, exact inference in such a model is intractable. Belief propagation can still be applied to a graph with cycles, however this loopy algorithm is not guaranteed to converge. Furthemore, if convergence does occur, any output from loopy belief propagation is approximate at best. In fact, it has been shown that the fixed points of loopy belief propagation correspond to the stationary points of the "Bethe free energy," a function that is often nonconvex [23], which may cause loopy belief propagation to converge to a nonglobal optimum.

We implement an alternative variational method, log-determinant relaxation [22], to run parameter estimation and inference in dense CRFs. Log-determinant relaxation (LDR) reframes the problem of computing marginal probabilities on graphs with cycles into a strictly convex optimization problem based on a Gaussian entropy bound in conjunction with both linear and semidefinite constraints on the marginal polytope. Under these constraints, efficient interior point methods can be used to approximate the log partition function, from which we can extract approximations of the node and edge likelihoods.

LDR is particularly well suited for CRF training. The convexity of the objective function provides the stability properties proven in [22]. Furthermore, the upper bound on the partition function translates into a lower bound on the conditional likelihood, guaranteeing that model approximation is controlled. Using LDR in both model estimation and inference, the dense CRF model also enjoys the benefit of error cancellation, which boosts overall classification performance [21]. At the time of writing, the application of LDR to discriminative models is novel.

Appendix A describes the inference algorithm for binary state nodes in a pairwise Ising model. The authors direct the reader to [22] for a more thorough treatment of the variational formulation and inference algorithm.

4 Experimental Results

We compare the three discriminative models on two different datasets: the CAL500 audio dataset and the PASCAL VOC2009 image dataset. In order to compare the quality of annotation across the three models, the area under the receiver operator characteristic (ROC) curve is computed for each tag. The ROC curve for a classifier plots the true positive rate, $tp/(tp+fn)$, against the false positive rate, $fp/(fp+tn)$, where tp is the number of true positives, fp the number of false positives, tn the number of true negatives, and fn the number of false negatives. The area under the curve (AUC) is a commonly used performance metric for classifiers. To compare performance over all tags, the mean AUC is computed for each model.

We also compare the accuracy of each classifier, defined as the number of correctly classified examples over the total number of examples. The mean accuracy is compared across models. [1]

4.1 Audio Tagging: CAL500 dataset

The CAL500 dataset is a corpus of 500 popular Western songs, each of which has been manually annotated by at least three human labelers using a distinct vocabulary to consider 135 musically relevant concepts: 29 instruments annotated as present or not; 22 vocal characteristics annotated as relevant or not; 36 genres were annotated as relevant or not; 18 emotions were rated on a scale from one to three (e.g. not happy, neutral, happy); 15 song concepts describing the acoustic qualities of the song, artist and recording; and 15 usage terms [18]. Using three separate human labelers for annotation captures some notion of agreement and defines our matrix $\mathbf{y}$. Figure 4 visualizes the co-occurrence matrices for instrument and genre labels.

Though the audio features of the CAL500 dataset are presented in a few different ways, we use the Mel-Frequency Cepstral Coefficient-delta (MFCC-delta) features, which capture the timbral evolution of three successive 23ms windows of audio, presented as 10,000 39-dimensional vectors for each song [7]. Employing the popular bag of features representation, these features are vector quantized and binned into counts such that each song is represented by a histogram of codewords. As in [7], the vector quantization process is as follows:

1. Standardize all feature vectors so they have mean 0 and standard deviation 1 in each dimension.
2. Run the k-means++ algorithm [1] on a subset of randomly selected feature vectors to find a set of K cluster centers
3. For each normalized feature vector f_{ji} in song j, assign that feature vector to the cluster k_{ji} with the smallest squared Euclidean distance.

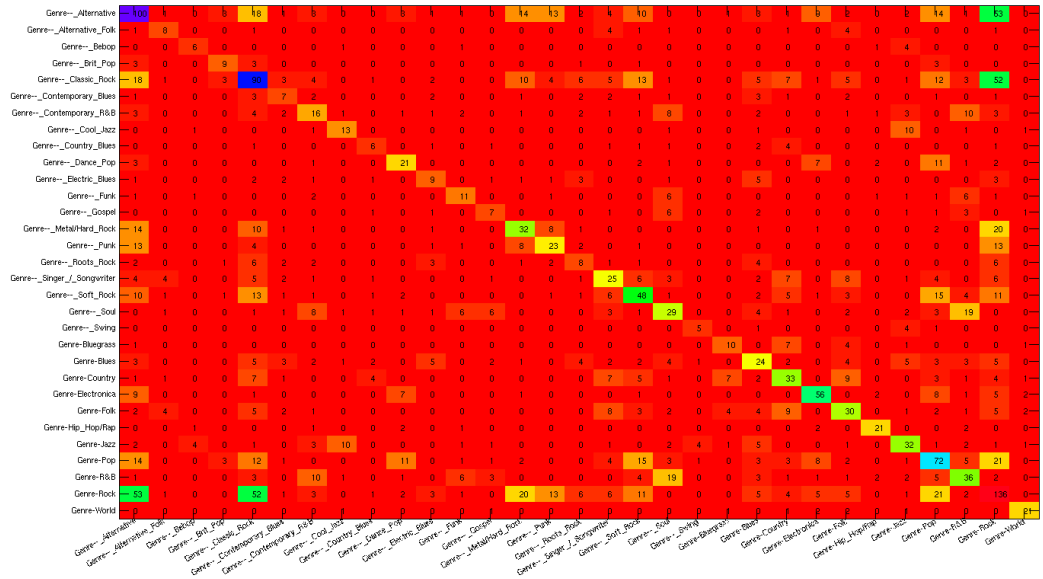
Vector quantizing the data allows us to represent each song j as a vector of codeword counts n_j such that

$$n_{jk} = \frac{1}{N_j} \sum_{i=1}^{N_j} \mathbf{1}(k_{ji}, k)$$

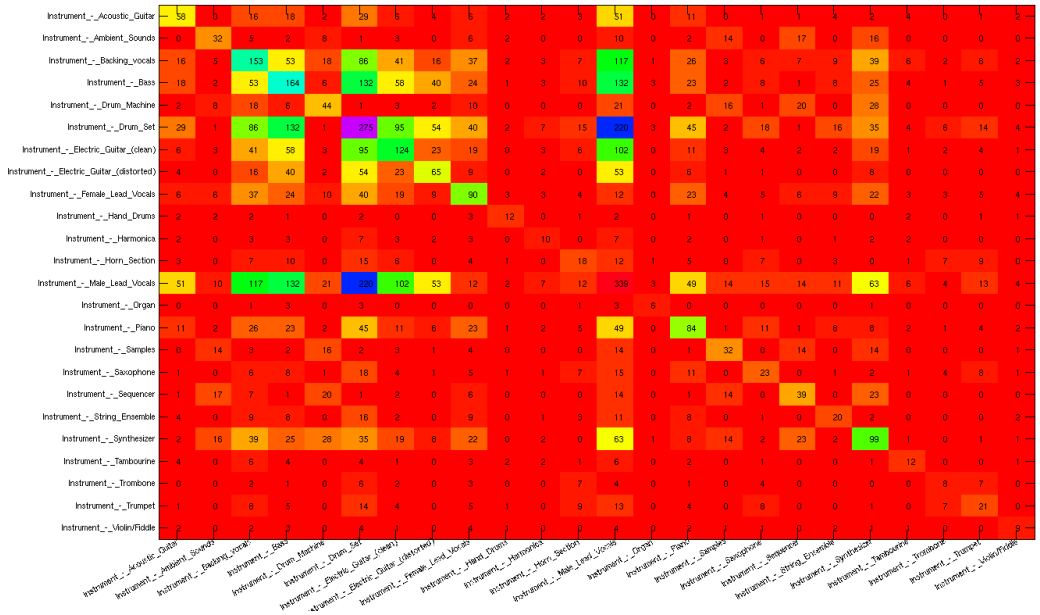
where N_j is the total number of feature vectors in song j, $\mathbf{1}(a, b)$ is the indicator function, returning 1 if $a = b$ and 0 otherwise. The value n_{jk} will be the proportion of feature vectors assigned to codeword k in song j.

Figure 5 compares the performance of the three different models on two subsets of tags. The first is the set of genre tags (visualized in figure 4a) and the second is the set of instrument tags (visualized in figure 4b). In both cases the model that performs best is the fully connected CRF running inference with LDR. Introducing context improves classification accuracy, while the tree-structured CRF sees a small dip in AUC. The improvement seen by the fully connected CRF may indicate that restricting the instrument tags to a tree structure may hurt classification accuracy, as some correlations may be unfairly represented. The average AUC improves by about 5 percent from the context free model to the fully connected model, while the average classification accuracy jumped by almost 3 percent.

[1] Models were implemented in Matlab using Mark Schmidt's "minFunc" and "UGM" packages [12, 13].



(a) Context matrix for the genre labels of the CAL500 dataset.



(b) Context matrix for the instrument labels of the CAL500 dataset.

Figure 4: CAL500 context matrices.

4.2 Image Tagging: PASCAL VOC2009 dataset

We apply these three models to the PASCAL Visual Object Classes Challenge 2009 (VOC2009) dataset [6]. The training and validation data consist of 7,054 images with 20 object categories labeled. These categories are organized by object class as follows:

- Person: person
- Animal: bird, cat, cow, dog, horse, sheep
- Vehicle: aeroplane, bicycle, boat, bus, car, motorbike, train

- Indoor: bottle, chair, dining table, potted plant, sofa, tv/monitor

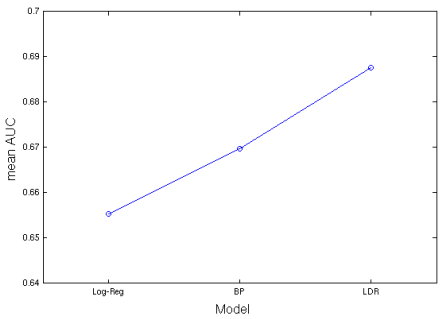
Figure 6 visualizes the co-occurrence counts for each object class.

Each image is reduced to a set of sparse scale-invariant feature transforms (SIFT) [8, 20] taking place at interest points. These SIFT features are vector quantized and reduced to a bag of codewords using the same process that produced the CAL500 features (however the SIFT features were not standardized before vector quantized).

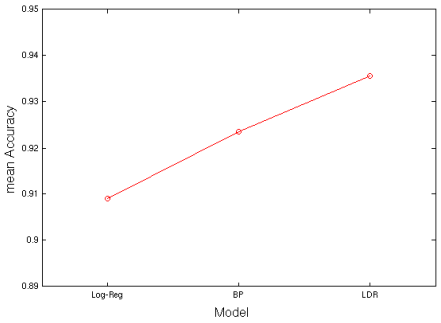
Table 1 compares the three methods on the vehicle set of object classes. Both the highest average AUC and accuracy are achieved by the fully connected model estimated and running inference with LDR. Both the tree structured and fully connected CRF present substantial improvements over independent logistic regression. Figure 7 compares the average performance of the three models over all 20 tags. Over all tags, the average AUC increases by over 13 percent from the independent to the fully connected model, while the average accuracy increases 5 percent.

5 Discussion

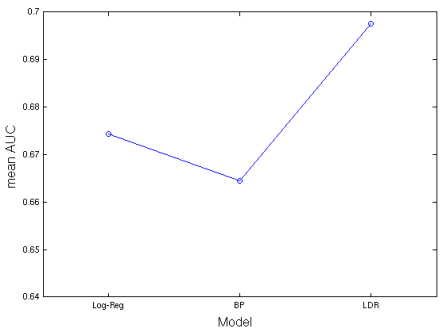
It naturally follows that introducing context to the task of annotation will be beneficial; however, highly correlated models often present a computational problem. This paper introduces two computationally tractable ways to introduce varying levels of context to discriminative annotation models. Both methods show substantial improvement over independent models. The authors also speculate that all three models could be further improved by a richer feature representation. For example, seg-



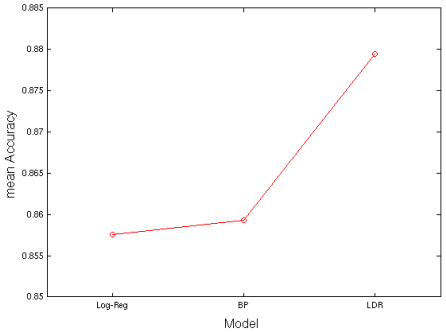
(a) Average AUC by model, genre tags.



(b) Average Accuracy by model, genre tags.



(c) Average AUC by model, instrument tags.



(d) Average Accuracy by model, instrument tags.

Figure 5: Summary of CAL500 results (K=500 codewords): top row shows genre related tags and bottom row shows instrument related tags. Each plot is a comparison of the three models: independent (Log-Reg), tree-structured (BP) and fully connected (LDR) CRFs. We ran a tenfold cross validation experiment, splitting our data into 10 disjoint 50-song sets at random.

Table 1: Model comparison by transportation tag: independent (REG), tree-structured (BP), and dense (LDR) CRFs. Features used are SIFT interest points (K=500). Node features consist of the bag of words, while edge features include only a bias term. Models are trained and tested on the subset of images that have at least one transportation object. All models are regularized.

	aeroplane	bicycle	boat	bus	car	motorbike	train	mean
AUCs:								
REG:	0.7650	0.6830	0.6489	0.7058	0.5388	0.6500	0.6097	0.6573
BP:	0.8241	0.7301	0.6785	0.7883	0.5744	0.6964	0.6759	0.7097
LDR:	0.8549	0.7900	0.7518	0.7841	0.6230	0.7348	0.7044	0.7490
Accuracy:								
REG:	0.8339	0.7736	0.7901	0.8281	0.5942	0.7438	0.7653	0.7613
BP:	0.8612	0.8463	0.8463	0.8818	0.6388	0.8314	0.8347	0.8201
LDR	0.8661	0.8579	0.8744	0.8950	0.6959	0.8529	0.8636	0.8437

menting an image before feature extraction may reduce some background noise, resulting in more accurate learning and inference.

In this paper, co-occurrence serves as the only source of context. In other contextual annotation models [11], other sources of context are used, namely Google Sets. Co-occurrence counts are naturally biased, as they reflect only the empirical probability of two objects appearing together in the dataset. Seeking other sources of context may further improve classification performance when examining unseen pairs of objects. Modeling the hierarchical context may also improve models. For example, first determining if a scene is indoors or outdoors (or determining if a song is performed by a rock group or an orchestra) may provide valuable context before individual annotations are classified. It would also be interesting to see the effect of contextual models on labelling partially labeled data. For example, if an image is known to have a car in it, how are the likelihood of other labels affected. An independent model would see no benefit, whereas a well trained contextual model should see improvement.

Also, using a different regularization scheme may improve model results. All models are implemented with L_2 regularization (with varying values of λ). In estimating a fully connected CRF, L_1 regularization may provide a sort of model selection, as edge parameters would more quickly go to

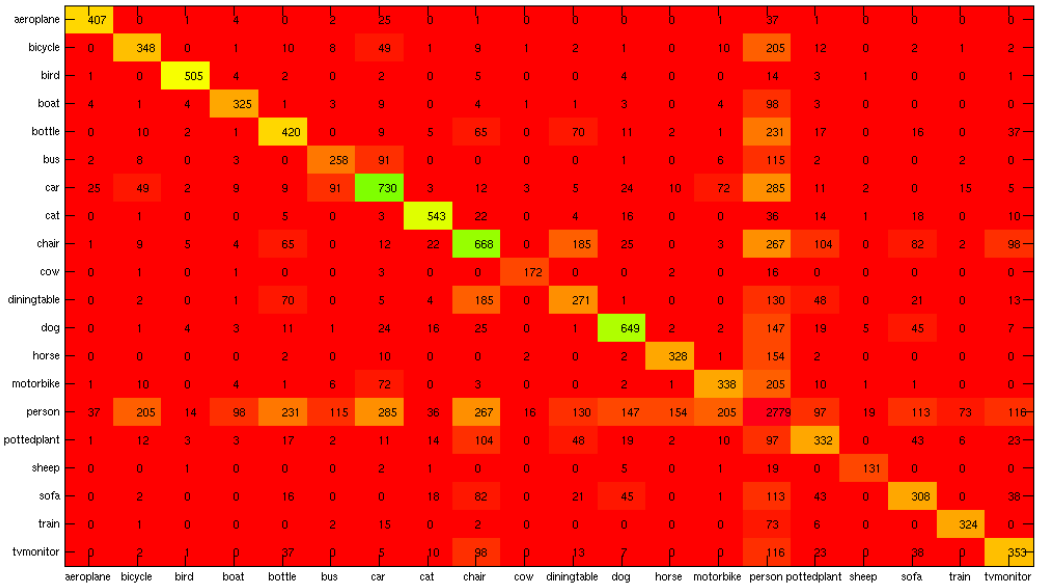
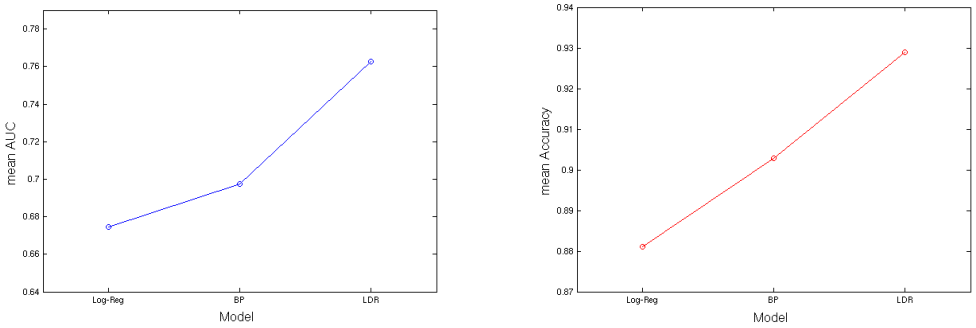


Figure 6: Context matrix for the PASCAL VOC2009 image dataset.



(a) Average AUC by model, all objects.

(b) Average Accuracy by model, all objects.

Figure 7: Summary of VOC2009 results: Each plot is a comparison of the three models: independent (Log-Reg), tree-structured (BP) and fully connected (LDR) CRFs. As context is introduced to the model, AUC and accuracy increase.

zero. As model size and complexity grows, it will be more necessary to study the effect of different regularization schemes.

Further work should also include a comparison of discriminative and generative models in both image and music annotation. The two datasets examined differ quite a bit in dimension. The CAL500 dataset has 500 examples with 135 labels, while the VOC2009 dataset has over 7,000 examples with only 20 labels. A smaller dataset may be more conducive to generative models, while a larger dataset may be more conducive to the discriminative CRFs presented in this paper [10]. Furthermore, in these discriminative models, as the dataset grows so does training time. Using stochastic gradient methods may speed up training in a real world system.

6 Acknowledgements

I'd like to thank my advisor, Erik Sudderth, for his infinite patience and endless stream of ideas and intuition that were invaluable to both this research project and my fundamental understanding of machine learning and statistics. Erik encouraged me to explore the marriage of my two interests, music and computer science, and this computer audition (and vision) project is the result.

Appendix

A Log-Determinant Relaxation

To clarify LDR, consider a binary MRF $p(y;\theta)$ over the vector $Y \in \{-1,1\}^n$, described by the graph structure $G = (V,E)$ and the pairwise Ising model

$$p(y;\theta) = \exp\left\{\sum_{s\in V}\theta_s y_s + \sum_{(s,t)\in E}\theta_{st}y_s y_t - A(\theta)\right\} \tag{13}$$

with

$$A(\theta) = \sum_{y\in Y^n}\exp\left\{\sum_{s\in V}\theta_s y_s + \sum_{(s,t)\in E}\theta_{st}y_s y_t\right\}. \tag{14}$$

Define the marginal polytope $MARG(G;\phi)$. This can be thought of as the convex set of all distributions over Y^n whose vertices correspond to the distribution that puts all of its mass on one arrangement of vector y. In the case of $y_s \in \{-1, 1\}$ this set will have 2^n vertices, and is defined as

$$MARG(G;\phi) := \{\mu \in \mathbb{R}^n | \sum_{y \in Y^n} \phi(y)p(y) = \mu \text{ for some distribution } p\}. \tag{15}$$

Let $\mu_s = \mathbb{E}[y_s]$ and $\mu_{st} = \mathbb{E}[y_s y_t]$ be the associated moments of the spin vector. It can be shown that an upper bound on the log partition takes the form

$$A(\theta) \leq \sup_{\mu \in MARG(G;\phi)} \{\langle \theta, \mu \rangle + H(p(y|\theta(\mu)))\} \tag{16}$$

where $\langle \cdot \rangle$ denotes inner product, and $H(p(y|\theta(\mu)))$ is the entropy of $p(y;\theta)$ when θ parameterizes a distribution that reflects the expected values determined by μ. Let $M_1[\mu]$ be a second-order moment matrix that has the form

$$M_1[\mu] = \begin{bmatrix} 1 & \mu_1 & \mu_2 & ... & \mu_{n-1} & \mu_n \\ \mu_1 & 1 & \mu_{12} & ... & ... & \mu_{1n} \\ \mu_2 & \mu_{21} & 1 & ... & ... & \mu_{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mu_{n-1} & \vdots & \vdots & \vdots & \vdots & \mu_{(n-1)n} \\ \mu_n & \mu_{n1} & \mu_{n2} & ... & \mu_{n(n-1)} & 1 \end{bmatrix}. \tag{17}$$

It can be shown that the binary marginal polytope $MARG(G)$ is contained within the semidefinite constraint set $SDEF_1(G) := \{\mu \in \mathbb{R}^n | M_1[\mu] \succeq 0\}$. Furthermore, there exist linear constraints that any member μ of the marginal polytope must satisfy. These are derived from the probabilistic interpretation that μ must abide by, such that $\mu_s = \mathbb{E}[Y_s] = 2Pr(y_s = 1) - 1$ for the spin representation of y.

Furthermore, it is known that the differential entropy of any continuous random vector $\tilde{Y}$ is upper bounded by the differential entropy of a Gaussian with a matched covariance matrix. With this entropy bound, it can be shown that for any compact convex outer bound $OUT(G)$ on the marginal polytope $MARG(G)$, the log partition function $A(\theta)$ is upper bounded by the solution to the following variational problem:

$$A(\theta) \leq \max_{\mu \in OUT(G), M_1[\mu] \succeq 0} \left\{ \langle \theta, \mu \rangle + \frac{1}{2} \log\det \left[M_1(\mu) + \frac{1}{3} blkdiag[0, I_n] \right] \right\} + \frac{n}{2} \log(2\pi e) \tag{18}$$

where $blkdiag[0, I_n]$ is an $(n+1) \times (n+1)$ block-diagonal matrix with a 1×1 zero block, and another $n \times n$ identity block. The $\log\det[\cdot]$ term acts as a natural constraint to enforce $M_1[\mu] \succeq -\frac{1}{3} blkdiag[0, I_n]$, which is slightly weaker than $M_1[\mu] \succeq 0]$. A further relaxation yields

$$A(\theta) \leq \max_{\mu} \left\{ \langle \theta, \mu \rangle + \frac{1}{2} \log\det \left[M_1[\mu] + \frac{1}{3} blkdiag[0, I_n] \right] \right\}. \tag{19}$$

This relaxed optimization problem can be rewritten by introducing $Y := M_1[\mu] + \frac{1}{3} blkdiag[0, I_n]$. The weakened form the relaxation corresponds to $\max_{Y \succeq 0} \{\langle -A, Y \rangle + \log\det Y\}$ such that $diag(Y) = d$, where $d = [1, 4/3, ..., 4/3]^{\mathrm{T}}$, which can be solved efficiently by introducing Lagrange multipliers.

Using this relaxed optimization problem to uncover the optimal matrix, the algorithm to obtain marginal beliefs is as follows:

1. Construct a matrix A such that $\langle -A, Y \rangle = 2\sum_{s \in V}(\theta_s \mu_s) + 2\sum_{(s,t) \in E}(\theta_{st}\mu_{st})$, that is

$$A = -\begin{bmatrix} 0 & \theta_1 & \theta_2 & ... & \theta_{n-1} & \theta_n \\ \theta_1 & 0 & \theta_{12} & ... & ... & \theta_{1n} \\ \theta_2 & \theta_{21} & 0 & ... & ... & \theta_{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \theta_{n-1} & \vdots & \vdots & \vdots & \vdots & \theta_{(n-1)n} \\ \theta_n & \theta_{n1} & \theta_{n2} & ... & \theta_{n(n-1)} & 0 \end{bmatrix}. \tag{20}$$

2. Initialize Lagrange multipliers $\lambda \in \mathbb{R}^{n+1}$ such that $A + diag(\lambda)$ is positive definite (i.e. diagonally dominant).
3. Find the λ^* that minimizes the following Lagrangian dual function

$$Q(\lambda) = -(n+1) - \log\det[A + diag(\lambda)] + \langle diag(\lambda), diag(d) \rangle \tag{21}$$

where its gradient and Hessian take the form

$$\nabla Q(\lambda) = -diag[A + diag(\lambda)]^{-1} + d \tag{22}$$

$$\nabla^2 Q(\lambda) = [A + diag(\lambda)]^{-1} \odot [A + diag(\lambda)]^{-1} \tag{23}$$

and $\odot$ denotes Hadamard product.

4. The optimal value of Y is given by $Y^* = (A + diag(\lambda^*))^{-1}$, and the optimal moment matrix is given by $M_1[\mu^*] = Y^* - \frac{1}{3} blk diag[0, I_n]$.
5. The node and edge beliefs can be extracted from the entries in moment matrix, μ^*, and an upper bound on the partition function follows from equation 19.

Again, the authors direct the reader to [22] for a more thorough treatment of LDR.

References

[1] D. Arthur and S. Vassilvitskii. k-means++: The advantages of careful seeding. *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035, 2007.

[2] T. Bertin-Mahieux, D. Eck, F. Maillet, and P. Lamere. Autotagger: a model for predicting social tags from acoustic features on large music databases. *Journal of New Music Research*, 37(2):115–135, 2008.

[3] I. Biederman, R. Mezzanotte, and J. Rabinowitz. Scene perception: detecting and judging objects undergoing relational violations. *Cognitive Psychology*, 14(2):143–177, 1982.

[4] D. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

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

[6] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2009 (VOC2009) Results. `http://www.pascal-network.org/challenges/VOC/voc2009/workshop/index.html`.

[7] M. Hoffman, D. Blei, and P. Cook. Easy as cba: A simple probabilistic model for tagging music. *In ISMIR*, 2009.

[8] David G. Lowe. Object recognition from local scale-invariant features. *Proceedings of the International Conference on Computer Vision*, 2:1150–1157, 1999.

[9] M. Mandel and D. Ellis. Labrosa's audio classification submissions. *mirex*, 2008.

[10] A. Y. Ng and M. I. Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. *Advances in Neural Information Processing Systems (NIPS)*, 14, 2002.

[11] A. Rabinovich, A. Vedaldi, C Galleguillos, E. Wiewiora, and S. Belongie. Objects in context. 2007.

[12] Mark Schmidt. minfunc software package. `http://www.cs.ubc.ca/~schmidtm/Software/minFunc.html`, 2005.

[13] Mark Schmidt. ugm software package. `http://www.cs.ubc.ca/~schmidtm/Software/UGM.html`, 2007.

[14] C. Sutton and A. McCallum. An introduction to conditional random fields for relational learning. *In ISMIR*, 2009.

[15] K. Trohidis, G. Tsoumakas, G. Kalliris, and I. Vlahavas. Multilabel classification of music into emotions. *In Proceedings of the 9th International Conference on Music Information Retrieval (ISMIR)*, 2008.

[16] D. Turnbull, L. Barrington, and G. Lanckriet. Modeling music and words using a multi-class naive bayes approach. *Introduction to Statistical Relational Learning*, 2006.

[17] D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet. Exploring the semantic annotation and retrieval of sound. *Technical Report CAL-2007-01, San Diego*, 2007.

[18] D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet. Towards musical query-by-semantic-description using the cal500 data set. *In Proc. ACM SIGIR*, pages 439–446, 2007.

[19] D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet. Semantic annotation and retrieval of music and sound effects. *IEEE Transactions on Audio Speech and Language Processing*, 2008.

[20] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. `http://www.vlfeat.org/`, 2008.

[21] M. Wainwright. Estimating the 'wrong' graphical model: Benefits in the computation-limited setting. *Journal of Machine Learning Research*, 7:1829–1859, 2006.

[22] M. Wainwright and M. Jordan. Log determinant relaxation for approximate inference in discrete markov random fields. *IEEE Transactions on Signal Processing*, 54 (6):1–11, 2006.

[23] J. Yedidia, W.T. Freeman, and Y. Weiss. Constructing free energy approximations and generalized belief propagation algorithms. *IEEE Transactions on Information Theory*, 51:2282–2312, 2005.