Laplace Maximum Margin Markov Networks

Jun Zhu[†*] JUN-ZHU@MAILS.TSINGHUA.EDU.CN
Eric P. Xing[†] EPXING@CS.CMU.EDU
Bo Zhang[*] DCSZB@MAIL.TSINGHUA.EDU.CN

[*]Department of Computer Science and Technology, Tsinghua University, Beijing 100084 China
[†]School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213 USA

Abstract

We propose Laplace max-margin Markov networks (LapM^{3}N), and a general class of Bayesian M^{3}N (BM^{3}N) of which the LapM^{3}N is a special case with sparse structural bias, for robust structured prediction. BM^{3}N generalizes extant structured prediction rules based on point estimator to a Bayes-predictor using a learnt distribution of rules. We present a novel *Structured Maximum Entropy Discrimination* (SMED) formalism for combining Bayesian and max-margin learning of Markov networks for structured prediction, and our approach subsumes the conventional M^{3}N as a special case. An efficient learning algorithm based on variational inference and standard convex-optimization solvers for M^{3}N, and a generalization bound are offered. Our method outperforms competing ones on both synthetic and real OCR data.

1. Introduction

In recent years, log-linear models based on composite features that explicitly exploit the structural dependencies among elements in high-dimensional inputs (e.g., DNA strings, text sequences, image lattices) and structured interpretational outputs (e.g., gene segmentation, natural language parsing, scene description) have gained substantial popularity in learning structured predictions from complex data. Major instances of such models include the conditional random fields (CRFs) (Lafferty et al., 2001), Markov networks (MNs) (Taskar et al., 2003), and other specialized graphical models (Altun et al., 2003). Adding to the flexibilities and expressive power of such models, different learning paradigms have been explored, such as maximum likelihood estimation (Lafferty et al., 2001), and max-margin learning (Altun et al., 2003; Taskar et al., 2003; Tsochantaridis et al., 2004).

For domains with complex feature space, it is often desirable to pursue a "sparse" representation of the model that leaves out irrelevant features. Learning such a sparse model is key to reduce the rick of overfitting and achieve good generalizability. In likelihood-based estimation, sparse model fitting has been extensively studied. A commonly used strategy is to add an L_1-penalty to the likelihood function, which can also be viewed as a MAP estimation under a Laplace prior. Recent work along this line includes (Lee et al., 2006; Wainwright et al., 2006; Andrew & Gao, 2007).

This progress notwithstanding, little progress has been made so far on learning sparse MNs or log-linear models in general based on the max-margin principle, which is arguably a more desirable paradigm for training highly discriminative structured prediction models in a number of application contexts. While sparsity has been pursued in maximum margin learning of certain discriminative models such as SVM that are "unstructured" (i.e., with a univariate output), by using L_1-regularization (Bennett & Mangasarian, 1992) or by adding a cardinality constraint (Chan et al., 2007), generalization of these techniques to structured output space turns out to be extremely non-trivial. For example, although it appears possible to formulate sparse max-margin learning as a convex optimization problem as for SVM, both the primal and dual problems are hard to solve since there is no obvious way to exploit the conditional independence structures within a regularized MN to efficiently deal with the typically exponential number of margin constraints. Another empirical insight as we will show in this paper is that the L_1-regularized estimation is not so robust. Discarding the features that are not completely irrelevant can potentially hurt generalization ability.

In this paper, we propose a new formalism called *Structured Maximum Entropy Discrimination*

Appearing in *Proceedings of the 25th International Conference on Machine Learning*, Helsinki, Finland, 2008.

(SMED), which offers a general framework to combine Bayesian learning and max-margin learning of log-linear models for structured prediction. SMED is a generalization of the maximum entropy discrimination (Jaakkola et al., 1999) methods originally developed for classification to the broader problem of structured learning. It facilitates posterior inference of a full distribution of feature coefficients (i.e., weights), rather than a point-estimate as in the standard max-margin Markov network (M^3N) (Taskar et al., 2003), under a user-specified prior distribution of the coefficients and generalized maximum margin constraints. One can use the learned posterior distribution of coefficients to form a Bayesian max-margin Markov network (BM^3N) that is equivalent to a weighted sum of differentially parameterized M^3Ns, or one can obtain a MAP BM^3N. We show that, by using a Laplace prior for the feature coefficients, the resulting BM^3N is effectively a "sparse" max-margin Markov network, which we refer to as a Laplace M^3N ($LapM^3N$). But unlike the L_1-regularized maximum likelihood estimation, where sparsity is due to a hard threshold introduced by the Laplace prior (Kaban, 2007), the effect of Laplace prior in $LapM^3N$ is a biased posterior weighting of the parameters. Smaller parameters are shrunk more and thus robust estimation is achieved when the data have irrelevant features. The Bayesian formalism also makes the $LapM^3N$ less sensitive to regularization constants. Interestingly, a trivial assumption on the prior distribution of the coefficients, i.e., a standard (zero-mean and identity covariance) normal, reduces BM^3N to the standard M^3N, as shown in Theorem 3.

The paper is structured as follows. The next section reviews the basic structured prediction formalism and sets the stage for our model. Sec. 3 presents the SMED formalism and basic results on BM^3N. Sec. 4 presents $LapM^3N$ and a novel learning algorithm. Sec. 5 presents a generalization bound of BM^3N. Sec. 6 shows empirical results. Sec. 7 concludes this paper.

2. Preliminaries

Consider a structured prediction problem such as natural language parsing, image understanding, or DNA decoding. The objective is to learn a predictive function $h : \mathcal{X} \mapsto \mathcal{Y}$ from a structured input $\mathbf{x} \in \mathcal{X}$ (e.g., a sentence or an image) to a structured output $\mathbf{y} \in \mathcal{Y}$ (e.g., a sentence parsing or a scene annotation), where $\mathcal{Y} = \mathcal{Y}_1 \times \cdots \times \mathcal{Y}_l$ with $\mathcal{Y}_i = \{y_1, \ldots, y_{m_i}\}$ represents a combinatorial space of structured interpretations of multi-facet objects. For example, $\mathcal{Y}$ could correspond to the space of all possible instantiations of the part-of-speech (POS) tagging in the parse tree of a sentence, or the space of all possible ways of labeling entities over some segmentation of an image. The prediction $\mathbf{y} \equiv (y_1, \ldots, y_l)$ is *structured* because each individual label $y_i \in \mathcal{Y}_i$ within $\mathbf{y}$ must be determined in the context of other labels $y_{j \neq i}$, rather than independently as in a standard classification problem.

Let $F : \mathcal{X} \times \mathcal{Y} \mapsto \mathbb{R}$ represent a discriminant function over the input-output pairs from which one can define the predictive function h. A common choice of F is a linear model, which is based on a set of feature functions $f_k : \mathcal{X} \times \mathcal{Y} \mapsto \mathbb{R}$ and their weights w_k, i.e., $F(\mathbf{x}, \mathbf{y}; \mathbf{w}) = \mathbf{w}^\top \mathbf{f}(\mathbf{x}, \mathbf{y})$. Given F, the prediction function h is typically defined in terms of an optimization problem that maximizes F over the response variable $\mathbf{y}$ given input $\mathbf{x}$:

$$h_0(\mathbf{x}; \mathbf{w}) = \arg \max_{\mathbf{y} \in \mathcal{Y}(\mathbf{x})} F(\mathbf{x}, \mathbf{y}; \mathbf{w}). \quad (1)$$

Depending on the specific choice of the objective function $C(\mathbf{w})$ for estimating the parameter $\mathbf{w}$ (e.g., likelihood, or margin), incarnations of the general structured prediction formalism described above can be seen in models such as the CRFs (Lafferty et al., 2001), where $C(\mathbf{w})$ is the conditional likelihood of the true structured label; and the M^3N (Taskar et al., 2003), where $C(\mathbf{w})$ is the margin between the true label and any other label. Recent advances in structured prediction has introduced regularizations of $C(\mathbf{w})$ in the CRF context, so that a *sparse* $\mathbf{w}$ can be learned (Andrew & Gao, 2007). To the best of our knowledge, existing max-margin structured prediction methods utilize a single discriminant function $F(\,\cdot\,; \mathbf{w})$ defined by the "optimum" estimate of $\mathbf{w}$, similar to a practice in Frequentist statistics. In this paper, we propose a Bayesian version of the predictive rule in Eq. (1) so that the prediction function h can be obtained from a posterior mean over multiple (indeed infinitely many) $F(\,\cdot\,; \mathbf{w})$; and we also propose a new formalism and objective $C(\mathbf{w})$ that lead to a **Bayesian M^3N**, which subsumes the standard M^3N as a special case, and can achieve a posterior shrinkage effect on $\mathbf{w}$ that resembles L_1-regulatiztion. To our knowledge, although sparse graphical model learning based on various likelihood-based principles has recently received substantial attention (Lee et al., 2006; Wainwright et al., 2006), learning sparse networks based on the maximum margin principle has not yet been successfully explored. Our proposed method represents an initial foray in this important direction.

Before dwelling into exposition of the proposed approach, we end this section with a brief recapitulation of the basic M^3N that motivates this work, and provides a useful baseline that grounds the proposed approach. Under a max-margin framework, given training data $\mathcal{D} = \{\langle \mathbf{x}^i, \mathbf{y}^i \rangle\}_{i=1}^N$, we obtain a point estimate

of the weight vector $\mathbf{w}$ by solving the following max-margin problem P0 (Taskar et al., 2003):

$$\text{P0 (M}^3\text{N)}: \quad \min_{\mathbf{w},\xi} \ \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}\xi_i$$

$$\text{s.t. } \forall i, \forall \mathbf{y} \neq \mathbf{y}^i: \quad \mathbf{w}^\top \Delta\mathbf{f}_i(\mathbf{y}) \geq \Delta\ell_i(\mathbf{y}) - \xi_i,\ \xi_i \geq 0\ ,$$

where $\Delta\mathbf{f}_i(\mathbf{y}) = \mathbf{f}(\mathbf{x}^i, \mathbf{y}^i) - \mathbf{f}(\mathbf{x}^i, \mathbf{y})$ and $\mathbf{w}^\top\Delta\mathbf{f}_i(\mathbf{y})$ is the "margin" between the true label $\mathbf{y}^i$ and a prediction $\mathbf{y}$, $\Delta\ell_i(\mathbf{y})$ is a loss function with respect to $\mathbf{y}^i$, and ξ_i is a slack variable that absorbs errors in the training data. Various loss functions have been proposed in the literature (Tsochantaridis et al., 2004). In this paper, we adopt the *hamming loss* used in (Taskar et al., 2003): $\Delta\ell_i(\mathbf{y}) = \sum_{j=1}^{|\mathbf{x}^i|}\mathbb{I}(y_j \neq y_j^i)$, where $\mathbb{I}(\cdot)$ is an indicator function that equals to one if the argument is true and zero otherwise. The optimization problem P0 is intractable because the feasible space for $\mathbf{w}$, $\mathcal{F}_0 = \{\mathbf{w} : \mathbf{w}^\top\Delta\mathbf{f}_i(\mathbf{y}) \geq \Delta\ell_i(\mathbf{y}) - \xi_i;\ \forall i, \forall\mathbf{y} \neq \mathbf{y}^i\}$, is defined by $O(N|\mathcal{Y}|)$ number of constraints, and $\mathcal{Y}$ itself is exponential to the size of the input $\mathbf{x}$. Exploring sparse dependencies among individual labels y_i in $\mathbf{y}$, as reflected in the specific design of the feature functions (e.g., based on pair-wise labeling potentials), and convex duality of the objective, efficient algorithms based on cutting-plane (Tsochantaridis et al., 2004) or message-passing (Taskar et al., 2003) have been proposed to obtain an approximate optimum solution. As described shortly, these algorithms can be directly employed as subroutines in solving our proposed model.

3. Bayesian Maximum Margin Markov Networks

In this paper, we take a Bayesian approach and learn a distribution $p(\mathbf{w})$, rather than a point estimate of $\mathbf{w}$, in a max-margin manner. For prediction, we take the average over all the possible models, that is:

$$h_1(\mathbf{x}) = \arg\max_{\mathbf{y}\in\mathcal{Y}(\mathbf{x})} \int p(\mathbf{w})F(\mathbf{x},\mathbf{y};\mathbf{w})\,\mathrm{d}\mathbf{w}\,. \quad (2)$$

Now, the open question is how we can devise an appropriate objective function over $p(\mathbf{w})$, in a similar spirit as the L_2-norm cost over $\mathbf{w}$ in P0, that leads to an optimum estimate of $p(\mathbf{w})$. Below, we present a structured maximum entropy discrimination (SMED) framework that facilitates the estimation of a Bayesian M^{3}N defined by $p(\mathbf{w})$. As we show in the sequel, our Bayesian max-margin learning formalism offers several advantages like the PAC-Bayes generalization guarantee and estimation robustness.

3.1. SMED and the Bayesian M^{3}N

Given a training set $\mathcal{D}$, analogous to the feasible space $\mathcal{F}_0$ for weight vector $\mathbf{w}$ in an M^{3}N (i.e., problem P0), the feasible subspace $\mathcal{F}_1$ of weight distribution $p(\mathbf{w})$ is defined by a set of *expected* margin constraints:

$$\mathcal{F}_1 = \{p(\mathbf{w}) : \langle\Delta F_i(\mathbf{y};\mathbf{w}) - \Delta\ell_i(\mathbf{y})\rangle_{p(\mathbf{w})} \geq -\xi_i,\ \forall i, \mathbf{y} \neq \mathbf{y}^i\},$$

where $\Delta F_i(\mathbf{y};\mathbf{w}) = F(\mathbf{x}^i,\mathbf{y}^i;\mathbf{w}) - F(\mathbf{x}^i,\mathbf{y};\mathbf{w})$ and $\langle\cdot\rangle_p$ denotes the expectations with respect to p.

To choose the best distribution $p(\mathbf{w})$ from $\mathcal{F}_1$, the *maximum entropy principle* suggests that one can consider the distribution that minimizes its relative entropy with respect to some chosen prior p_0, as measured by the Kullback-Leibler divergence, $KL(p||p_0) = \langle\log(p/p_0)\rangle_p$. To accommodate the discriminative prediction problem we concern, instead of minimizing the usual KL, we optimize the generalized entropy (Dudík et al., 2007; Lebanon & Lafferty, 2001), or a regularized KL-divergence, $KL(p(\mathbf{w})||p_0(\mathbf{w})) + U(\xi)$, where $U(\xi)$ is a closed proper convex function over the slack variables. This leads to the following Structured Maximum Entropy Discrimination Model:

Definition 1 (The Structured Maximum Entropy Discrimination Model) *Given training data $\mathcal{D} = \{\langle\mathbf{x}^i,\mathbf{y}^i\rangle\}_{i=1}^N$, a discriminant function $F(\mathbf{x},\mathbf{y};\mathbf{w})$, a loss function $\Delta\ell_{\mathbf{x}}(\mathbf{y})$, and an ensuing feasible subspace $\mathcal{F}_1$ (defined above) for parameter distribution $p(\mathbf{w})$, the SMED model that leads to a prediction function of the form of Eq. (2) is defined by the following generalized relative entropy minimization with respect to a parameter prior $p_0(\mathbf{w})$:*

$$\text{P1}: \quad \min_{p(\mathbf{w}),\xi} \ KL(p(\mathbf{w})||p_0(\mathbf{w})) + U(\xi)$$
$$\text{s.t. } \ p(\mathbf{w}) \in \mathcal{F}_1,\ \xi_i \geq 0, \forall i.$$

The P1 defined above is a variational optimization problem over $p(\mathbf{w})$ in a subspace of valid parameter distributions. Since both the KL and the function U in P1 are convex, and the constraints in $\mathcal{F}_1$ are linear, P1 is a convex program, which can be solved via applying the calculus of variations to the Lagrangian to obtain a variational extremum, followed by a dual transformation of P1. Due to space limit, a detailed derivation is given in an extended version of this paper, and below we state the main results as a theorem.

Theorem 2 (Solution to SMED) *The variational optimization problem P1 underlying the SMED model gives rise to the following optimum distribution of Markov network parameters $\mathbf{w}$:*

$$p(\mathbf{w}) = \frac{1}{Z(\alpha)}p_0(\mathbf{w})\exp\{\sum_{i,\mathbf{y}}\alpha_i(\mathbf{y})[\Delta F_i(\mathbf{y};\mathbf{w}) - \Delta\ell_i(\mathbf{y})]\}, \quad (3)$$

where the Lagrangian multipliers $\alpha_i(\mathbf{y})$ (corresponding to constraints in $\mathcal{F}_1$) can be obtained by solving the dual problem of P1:

$$\text{D1}: \quad \max_{\alpha} \ -\log Z(\alpha) - U^\star(\alpha)$$
$$\text{s.t. } \ \alpha_i(\mathbf{y}) \geq 0,\ \forall i,\ \forall\mathbf{y},$$

where $U^\star(\cdot)$ represents the conjugate of the slack function $U(\cdot)$, i.e., $U^\star(\alpha) = \sup_\xi\big(\sum_{i,\mathrm{y}}\alpha_i(\mathrm{y})\xi_i - U(\xi)\big)$.

For a closed proper convex function $\phi(\mu)$, its conjugate is defined as $\phi^\star(\nu) = \sup_\mu[\nu^\top\mu - \phi(\mu)]$. In problem D1, by convex duality, the log normalizer $\log Z(\alpha)$ can be shown to be the conjugate of the KL-divergence. If the slack function is $U(\xi) = C\|\xi\| = C\sum_i \xi_i$, it is easy to show that $U^\star(\alpha) = \mathbb{I}_\infty(\sum_{\mathbf{y}} \alpha_i(\mathbf{y}) \leq C,\ \forall i)$, where $\mathbb{I}_\infty(\cdot)$ is a function that equals to zero when its argument holds true and infinity otherwise. Here, the inequality corresponds to the trivial solution $\xi = 0$, that is, the training data are perfectly separative. Ignoring this inequality does not affect the solution since the special case $\xi = 0$ is still included. Thus, the Lagrangian multipliers $\alpha_i(\mathbf{y})$ in the dual problem D1 comply with the set of constraints that $\sum_{\mathbf{y}} \alpha_i(\mathbf{y}) = C,\ \forall i$. Another example is $U(\xi) = KL(p(\xi)||p_0(\xi))$ by introducing uncertainty on the slack variables (Jaakkola et al., 1999). Some other U functions and their dual functions are studied in (Lebanon & Lafferty, 2001; Dudík et al., 2007).

The optimum parameter distribution $p(\mathbf{w})$ defined by Eq. (3), along with the predictive function $h_1(x;\mathbf{w})$ given by Eq. (2), jointly form what we would like to call a **Bayesian M^{3}N** (BM^{3}N). The close connection of BM^{3}N and M^{3}N is suggested by the striking isomorphisms of the opt-problem P1, the feasible space $\mathcal{F}_1$, and the predictive function h_1 underlying an BM^{3}N, to their counterparts P0, $\mathcal{F}_0$, and h_0, respectively, underlying an M^{3}N. Indeed, by making a special choice of a parameter prior in Eq. (3), based on the above discussion of conjugate functions in D1, we arrive at a reduction of D1 to an M^{3}N optimization problem. The following theorem makes this explicit.

Theorem 3 (Reduction of BM^{3}N to M^{3}N) *Assuming $F(\mathbf{x},\mathbf{y};\mathbf{w}) = \mathbf{w}^\top\mathbf{f}(\mathbf{x},\mathbf{y})$, $U(\xi) = \sum_i \xi_i$, and $p_0(\mathbf{w}) = \mathcal{N}(\mathbf{w}|0,I)$, where I denotes an identity matrix, then the Lagrangian multipliers $\alpha_i(\mathbf{y})$ are obtained by solving the following dual problem:*

$$\max_{\alpha} \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})\Delta\ell_i(\mathbf{y}) - \frac{1}{2}\|\sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})\Delta\mathbf{f}_i(\mathbf{y})\|^2$$
$$\text{s.t.} \sum_{\mathbf{y}} \alpha_i(\mathbf{y}) = C;\ \alpha_i(\mathbf{y}) \geq 0,\ \forall i,\ \forall \mathbf{y},$$

which, when applied to h_1, lead to a predictive function that is identical to $h_0(\mathbf{x};\mathbf{w})$ given by Eq. (1).

Proof: (sketch) Replacing $p_0(\mathbf{w})$ in Eq. (3) with $\mathcal{N}(\mathbf{w}|0,I)$, we can obtain the following closed-form expression of the $Z(\alpha)$ in $p(\mathbf{w})$:

$$\int \frac{1}{(2\pi)^{\frac{K}{2}}} \exp\{-\frac{\mathbf{w}^\top\mathbf{w}}{2} + \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})[\mathbf{w}^\top\Delta\mathbf{f}_i(\mathbf{y}) - \Delta\ell_i(\mathbf{y})]\}\,\mathrm{d}\mathbf{w}$$
$$= \exp\Big(-\sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})\Delta\ell_i(\mathbf{y}) + \frac{1}{2}\|\sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})\Delta\mathbf{f}_i(\mathbf{y})\|^2\Big).$$

As we have stated, the constraints $\sum_{\mathbf{y}} \alpha_i(\mathbf{y}) = C$ are due to the conjugate of $U(\xi) = \sum_i \xi_i$. □

Theorem 3 shows that in the supervised learning setting, M^{3}N is subsumed by the SMED model, and can be viewed as a special case of a Bayesian M^{3}N when the slack function is linear and the parameter prior is a standard normal. As described later, this connection renders many existing techniques for solving the M^{3}N directly applicable for solving the BM^{3}N. Note that although the distribution $p(\mathbf{w})$ in Eq. (3) has the same form as that of Bayesian CRFs (Qi et al., 2005), the underlying principles are fundamentally different.

Recent trend in pursuing "sparse" graphical models has led to the emergence of regularized version of CRFs (Andrew & Gao, 2007) and Markov networks (Lee et al., 2006; Wainwright et al., 2006). Interestingly, while such extensions have been successfully implemented by several authors in maximum likelihood learning of various sparse graphical models, they have not yet been explored in the context of maximum margin learning. Such a gap is not merely due to a negligence. Indeed, learning a sparse M^{3}N can be significantly harder as we discuss below.

As Theorem 3 reveals, an M^{3}N corresponds to a BM^{3}N with a standard normal prior for the weight vector $\mathbf{w}$. To encourage a sparse model, when using zero-mean normal prior, the weights of irrelevant features should peak around zero with very small variances. However, the isotropy of the variances in all dimensions in the standard normal prior makes M^{3}N infeasible to adjust the variances in different dimensions to fit sparse data. One way to learn a sparse model is to adopt the strategy of L_1-SVM to use L_1-norm instead of L_2-norm (a detailed description of this formulation and the duality derivation is available in the extended version of this paper). However, in both the primal and dual of an L_1-regularized M^{3}N, there is no obvious way to exploit the sparse dependencies among variables of the MN in order to efficiently deal with typically exponential number of constraints, which makes direct optimization or LP-formulation expensive. In this paper, we adopt the SMED framework that directly leads to a Bayesian M^{3}N, and employ a Laplace prior for $\mathbf{w}$ to learn a Laplace M^{3}N. When fitted to training data, the parameter posterior $p(\mathbf{w})$ under a Laplace M^{3}N has a shrinkage effect on small weights, which is similar to the L_1-regularizer in an M^{3}N. Although exact learning of a Laplace M^{3}N is still very hard, we show that it can be efficiently approximated by a variational inference procedure based on existing methods.

4. Laplace M^{3}N

The Laplace prior is $p_0(\mathbf{w}) = \prod_{k=1}^K \frac{\sqrt{\lambda}}{2} e^{-\sqrt{\lambda}|w_k|} = \big(\frac{\sqrt{\lambda}}{2}\big)^K e^{-\sqrt{\lambda}\|\mathbf{w}\|}$. The Laplace density is heavy tailed

and peaked at zero. Thus, it encodes the prior belief that the distribution of $\mathbf{w}$ is strongly peaked around zero. Another nice property is that the Laplace density is log-convex, which can be exploited to get convex estimation problems like LASSO (Tibshirani, 1996).

4.1. Variational Learning with Laplace Prior

Although in principle we have a closed-form solution of $p(\mathbf{w})$ in Theorem 2, the parameters $\alpha_i(\mathbf{y})$ are hard to estimate when using the Laplace prior. As we shall see in Section 4.2, exact integration will lead to a dual function that is difficult to maximize. Thus, we present a variational approximate learning approach.

Our approach is based on the hierarchical interpretation (Figueiredo, 2003) of the Laplace prior, that is, each w_k has a zero-mean Gaussian distribution $p(w_k|\tau_k) = \mathcal{N}(w_k|0, \tau_k)$ and the variance τ_k has an exponential hyper-prior density,

$$p(\tau_k|\lambda) = \frac{\lambda}{2}\exp\big\{-\frac{\lambda}{2}\tau_k\big\}, \quad \text{for } \tau_k \geq 0.$$

Let $p(\mathbf{w}|\tau) = \prod_{k=1}^K p(w_k|\tau_k)$, $p(\tau|\lambda) = \prod_{k=1}^K p(\tau_k|\lambda)$, then, $p_0(\mathbf{w}) = \int p(\mathbf{w}|\tau)p(\tau|\lambda)\,\mathrm{d}\tau$. Using the hierarchical representation and applying the Jensen's inequality, we get the following upper bound:

$$\begin{aligned} KL(p||p_0) &= -H(p) - \langle \log \int p(\mathbf{w}|\tau)p(\tau|\lambda)\,\mathrm{d}\tau \rangle_p \\ &\leq -H(p) - \langle \int q(\tau)\log\frac{p(\mathbf{w}|\tau)p(\tau|\lambda)}{q(\tau)}\,\mathrm{d}\tau \rangle_p \\ &\triangleq \mathcal{L}(p(\mathbf{w}), q(\tau)), \end{aligned}$$

where $q(\tau)$ is a variational distribution which is used to approximate $p(\tau|\lambda)$.

Substituting this upper bound for the KL in P1, we now solve the following problem,

$$\min_{p(\mathbf{w})\in\mathcal{F}_1; q(\tau); \xi} \mathcal{L}(p(\mathbf{w}), q(\tau)) + U(\xi). \tag{4}$$

This problem can be solved with an iterative minimization algorithm alternating between $p(\mathbf{w})$ and $q(\tau)$, as outlined in Algorithm 1, and detailed below.

Algorithm 1 Variational Bayesian Learning

Input: data $\mathcal{D} = \{\langle \mathbf{x}^i, \mathbf{y}^i\rangle\}_{i=1}^N$, constants C and λ, iteration number T
Output: posterior mean $\langle \mathbf{w}\rangle_p^T$
Initialize $\langle \mathbf{w}\rangle_p^1 \leftarrow 0$, $\Sigma_{\mathbf{w}}^1 \leftarrow I$
for $t = 1$ **to** $T-1$ **do**
  Step 1: solve (5) or (6) for $\langle \mathbf{w}\rangle_p^{t+1} = \Sigma_{\mathbf{w}}^t\eta$; update $\langle \mathbf{w}\mathbf{w}^\top\rangle_p^{t+1} \leftarrow \Sigma_{\mathbf{w}}^t + \langle \mathbf{w}\rangle_p^{t+1}(\langle \mathbf{w}\rangle_p^{t+1})^\top$.
  Step 2: use (7) to update $\Sigma_{\mathbf{w}}^{t+1} \leftarrow \mathrm{diag}(\sqrt{\frac{\langle w_k^2\rangle_p^{t+1}}{\lambda}})$.
end for

Step 1: Keep $q(\tau)$ fixed, we optimize (4) with respect to $p(\mathbf{w})$. Taking the same procedure as in solving P1, we get the posterior distribution $p(\mathbf{w})$ as follows,

$$\begin{aligned} p(\mathbf{w}) &\propto \exp\{\int q(\tau)\log p(\mathbf{w}|\tau)\,\mathrm{d}\tau - b\}\cdot\exp\{\mathbf{w}^\top\eta - L\} \\ &\propto \exp\{-\frac{1}{2}\mathbf{w}^\top\langle A^{-1}\rangle_q\mathbf{w} - b + \mathbf{w}^\top\eta - L\} \\ &= \mathcal{N}(\mathbf{w}|\mu_{\mathbf{w}}, \Sigma_{\mathbf{w}}), \end{aligned}$$

where $\eta = \sum_{i,\mathbf{y}}\alpha_i(\mathbf{y})\Delta\mathbf{f}_i(\mathbf{y})$, $L = \sum_{i,\mathbf{y}}\alpha_i(\mathbf{y})\Delta\ell_i(\mathbf{y})$, $A = \mathrm{diag}(\tau_k)$, and $b = KL(q(\tau)||p(\tau|\lambda))$ is a constant. The posterior mean and variance are $\langle \mathbf{w}\rangle_p = \mu_{\mathbf{w}} = \Sigma_{\mathbf{w}}\eta$ and $\Sigma_{\mathbf{w}} = (\langle A^{-1}\rangle_q)^{-1} = \langle \mathbf{w}\mathbf{w}^\top\rangle_p - \langle \mathbf{w}\rangle_p\langle \mathbf{w}\rangle_p^\top$, respectively. The dual parameters α are estimated by solving the following dual problem:

$$\begin{aligned} \max_{\alpha}\ & \sum_{i,\mathbf{y}}\alpha_i(\mathbf{y})\Delta\ell_i(\mathbf{y}) - \frac{1}{2}\eta^\top\Sigma_{\mathbf{w}}\eta \\ \text{s.t.}\ & \sum_{\mathbf{y}}\alpha_i(\mathbf{y}) = C;\ \alpha_i(\mathbf{y}) \geq 0,\ \forall i,\ \forall\mathbf{y}. \end{aligned} \tag{5}$$

This dual problem can be directly solved using existing algorithms developed for M^{3}N, such as (Taskar et al., 2003; Bartlett et al., 2004). Alternatively, we can solve the following primal problem:

$$\begin{aligned} \min_{\mathbf{w},\xi}\ & \frac{1}{2}\mathbf{w}^\top\Sigma_{\mathbf{w}}^{-1}\mathbf{w} + C\sum_{i=1}^N\xi_i \\ \text{s.t.}\ & \mathbf{w}^\top\Delta\mathbf{f}_i(\mathbf{y}) \geq \Delta\ell_i(\mathbf{y}) - \xi_i;\ \xi_i \geq 0,\ \ \forall i,\ \forall\mathbf{y} \neq \mathbf{y}^i. \end{aligned} \tag{6}$$

It is easy to show that the solution of problem (6) leads to the posterior mean of $\mathbf{w}$ under $p(\mathbf{w})$. The primal problem can be solved with subgradient (Ratliff et al., 2007) or extragradient (Taskar et al., 2006) methods.

Step 2: Keep $p(\mathbf{w})$ fixed, we optimize (4) with respect to $q(\tau)$. Take the derivative of $\mathcal{L}$ with respect to $q(\tau)$ and set it to zero, then we get $q(\tau) = \prod_{k=1}^K q(\tau_k)$. Each $q(\tau_k)$ is computed as follows:

$$\begin{aligned} \forall k: \quad q(\tau_k) &\propto p(\tau_k|\lambda)\exp\big\{\langle \log p(w_k|\tau_k)\rangle_p\big\} \\ &\propto \mathcal{N}(\sqrt{\langle w_k^2\rangle_p}|0,\tau_k)\exp(-\frac{1}{2}\lambda\tau_k). \end{aligned}$$

The normalization factor is $\int\mathcal{N}(\sqrt{\langle w_k^2\rangle_p}|0,\tau_k)\cdot\frac{\lambda}{2}\exp(-\frac{1}{2}\lambda\tau_k)\,\mathrm{d}\tau_k = \frac{\sqrt{\lambda}}{2}\exp(-\sqrt{\lambda\langle w_k^2\rangle_p})$. The expectations $\langle\tau_k^{-1}\rangle_q$ required in calculating $\langle A^{-1}\rangle_q$ are calculated as follows,

$$\langle\frac{1}{\tau_k}\rangle_q = \int\frac{1}{\tau_k}q(\tau_k)\,\mathrm{d}\tau_k = \sqrt{\frac{\lambda}{\langle w_k^2\rangle_p}}. \tag{7}$$

We iterate between the above two steps until convergence. Then, we use the posterior distribution $p(\mathbf{w})$ to make prediction. For irrelevant features, the variances should converge to zeros and thus lead to a sparse estimation. The intuition behind this iterative minimization algorithm is as follows. First, we use a Gaussian distribution to approximate the Laplace distribution and thus get a QP problem that is analogous to that of M^{3}N; then, the second step updates the covariance matrix in the QP problem with an exponential hyper-prior on the variance.

4.2. Insights

To see how the Laplace prior affects the posterior distribution, we do the following calculations. Substitute the hierarchical representation of the Laplace prior into $p(\mathbf{w})$ in Theorem 2, and we get:

$$\begin{aligned} Z(\alpha) &= \int\int p(\mathbf{w}|\tau)p(\tau|\lambda)\,\mathrm{d}\tau \cdot \exp\{\mathbf{w}^\top\eta - L\}\,\mathrm{d}\mathbf{w} \\ &= \int p(\tau|\lambda)\int p(\mathbf{w}|\tau)\cdot\exp\{\mathbf{w}^\top\eta - L\}\,\mathrm{d}\mathbf{w}\;\mathrm{d}\tau \\ &= \exp\{-L\}\prod_{k=1}^{K}\frac{\lambda}{\lambda-\eta_k^2}, \end{aligned} \tag{8}$$

where $\eta_k = \sum_{i,\mathbf{y}}\alpha_i(\mathbf{y})(f_k(\mathbf{x}^i,\mathbf{y}^i) - f_k(\mathbf{x}^i,\mathbf{y}))$ and an additional constraint is $\forall k,\ \eta_k^2 < \lambda$. Otherwise, the integration is infinity. Using the result (8), we can get:

$$\frac{\partial \log Z}{\partial \alpha_i(\mathbf{y})} = \mu^\top\Delta\mathbf{f}_i(\mathbf{y}) - \Delta\ell_i(\mathbf{y}), \tag{9}$$

where μ is a column vector and $\mu_k = \frac{2\eta_k}{\lambda-\eta_k^2}$, $\forall 1 \le k \le K$. An alternative way is using the definition of Z : $Z = \int p_0(\mathbf{w})\cdot\exp\{\mathbf{w}^\top\eta - L\}\,\mathrm{d}\mathbf{w}$. We can get:

$$\frac{\partial \log Z}{\partial \alpha_i(\mathbf{y})} = \langle\mathbf{w}\rangle_p^\top\Delta\mathbf{f}_i(\mathbf{y}) - \Delta\ell_i(\mathbf{y}). \tag{10}$$

Comparing Eqs. (9) and (10), we get $\langle\mathbf{w}\rangle_p = \mu$, that is, $\langle w_k\rangle_p = \frac{2\eta_k}{\lambda-\eta_k^2}$, $\forall 1 \le k \le K$. Similar calculation can lead to the result that in M^{3}N (standard normal prior) $\langle\mathbf{w}\rangle_p = \eta$. Figure 1 shows the posterior means (any dimension) when the priors are standard normal, Laplace with $\lambda = 4$, and Laplace with $\lambda = 6$. We can see that with a Laplace prior, the parameters are shrunk around zero. The larger the λ value is, the greater the shrinkage effect. For a fixed λ, the shape of the posterior mean is smoothly nonlinear but no component is explicitly discarded, that is, no weight is set to zero. This is different from the shape of a L_1-regularized maximum likelihood estimation (Kaban, 2007) where an interval exists around the origin and parameters falling into this interval are set to zeros.

Note that if we use the exact integration as in Eq. (8), the dual problem D1 will maximize $L - \sum_{k=1}^{K}\log\frac{\lambda}{\lambda-\eta_k^2}$. Since η_k^2 appears within a logarithm, the optimization problem would be very hard to solve. Thus, we turn to a variational approximation method.

5. Generalization bound

The PAC-Bayes bound (Langford et al., 2001) provides a theoretical motivation to learn an averaging model as in P1 which minimizes the KL-divergence and simultaneously satisfies the discriminative classification constraints. To apply it to our structured learning setting, we assume that the discriminant functions are bounded, that is, $F \in \mathcal{H} : \mathcal{X}\times\mathcal{Y} \to [-c, c]$ for all $\mathbf{w}$,

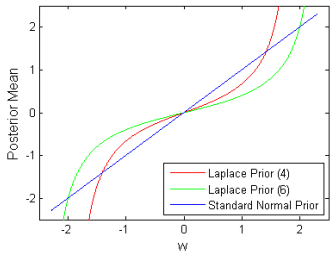

Figure 1. Posterior mean with different priors against the estimation of M^{3}N (i.e. with the standard normal prior).

where c is a positive constant. Recall that our averaging model is $h(\mathbf{x},\mathbf{y}) = \langle F(\mathbf{x},\mathbf{y};\mathbf{w})\rangle_{p(\mathbf{w})}$. We define the margin of an example $(\mathbf{x},\mathbf{y})$ for such a function h as $M(h,\mathbf{x},\mathbf{y}) = h(\mathbf{x},\mathbf{y}) - \max_{\mathbf{y}'\neq\mathbf{y}} h(\mathbf{x},\mathbf{y}')$. Clearly, the model h makes a wrong prediction on $(\mathbf{x},\mathbf{y})$ only if $M(h,\mathbf{x},\mathbf{y}) \le 0$. Let Q be a distribution over $\mathcal{X}\times\mathcal{Y}$, and let $\mathcal{D}$ be a sample of N examples randomly drawn from Q. We have the following PAC-Bayes theorem.

Theorem 4 (PAC-Bayes Bound of BM^{3}N) *Let p_0 be any continuous probability distribution over $\mathcal{H}$ and let $\delta \in (0,1)$. If $F \in \mathcal{H} : \mathcal{X}\times\mathcal{Y} \to [-c,c]$ for all $\mathbf{w}$, then with probability at least $1-\delta$ over random samples $\mathcal{D}$ of Q, for very distribution p over $\mathcal{H}$ and for all margin thresholds $\gamma > 0$:*

$$\begin{aligned} &\Pr_Q(M(h,\mathbf{x},\mathbf{y}) \le 0) \le \Pr_{\mathcal{D}}(M(h,\mathbf{x},\mathbf{y}) \le \gamma) \\ &\quad + O\left(\sqrt{\frac{\gamma^{-2}KL(p||p_0)\ln(N|\mathcal{Y}|) + \ln N + \ln\delta^{-1}}{N}}\right). \end{aligned}$$

Here, $\Pr_Q(.)$ stands for $\langle.\rangle_Q$ and $\Pr_{\mathcal{D}}(.)$ stands for the empirical average on $\mathcal{D}$. The proof follows the same structure as the original PAC-Bayes bound proof, with consideration of the margins. Due to space limit, details of the proof are given in the extended paper.

6. Experiments

In this section, we present some empirical results of LapM^{3}N on both synthetic and real data sets. We compare LapM^{3}N with M^{3}N, CRFs, L_1-regularized CRFs (L_1-CRFs), and L_2-regularized CRFs (L_2-CRFs). We use the quasi-Newton method (Andrew & Gao, 2007) to learn L_1-CRFs.

6.1. Synthetic Data Sets

6.1.1. I.I.D Features

The first experiment is conducted on synthetic sequence data with 100 i.i.d features. We generate three types of data sets with 10, 30, and 50 relevant features. For each setting, we randomly generate 10 linear-chain CRFs with 8 binary labeling states. The feature functions include: a real valued state-feature function over a one dimensional input feature and a class label; and

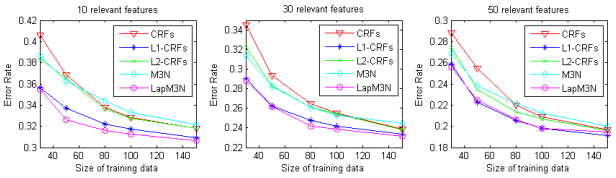

Figure 2. Evaluation results on data sets with i.i.d features.

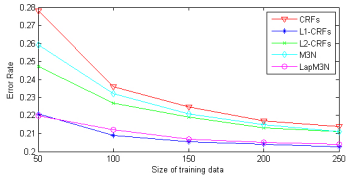

Figure 3. Results on data sets with 30 relevant features.

4 (2×2) binary transition-feature functions capturing pairwise label dependencies. For each model we generate a data set of 1000 samples. For each sample, we first *independently* draw the 100 features from a standard normal distribution, and then apply a Gibbs sampler to assign a label sequence with 5000 iterations.

For each data set, we randomly draw a part as training data and use the rest for testing. The numbers of training data are 30, 50, 80, 100, and 150. The QP problem is solved with the exponentiated gradient method (Bartlett et al., 2004). In all the following experiments, the regularization constant of L_1-CRFs and L_2-CRFs is chosen from $\{0.01, 0.1, 1, 4, 9, 16\}$ by a 5-fold cross-validation in training. For LapM^{3}N, we use the same method to choose λ from 20 roughly evenly spaced values between 1 and 268. For each setting, the average over 10 data sets is the final performance.

The results are shown in Figure 2. All the results of LapM^{3}N are achieved with 3 iterations of the variational learning. Under different settings LapM^{3}N consistently outperforms M^{3}N and performs comparably with L_1-CRFs. But note that the synthetic data come from simulated CRFs. Both L_1-CRFs and L_2-CRFs outperform the un-regularized CRFs. One interesting result is that M^{3}N and L_2-CRFs perform comparably. This is reasonable because as derived by Lebanon and Lafferty (2001) and noted by Globerson et al. (2007) the L_2-regularized MLE of CRFs has a similar convex dual as that of M^{3}N. The only difference is the loss they try to optimize. CRFs optimize the log-loss while M^{3}N optimizes the hinge-loss. As the number of training data increase, all the algorithms consistently get higher performance. The advantage of LapM^{3}N is more obvious when there are fewer relevant features.

6.1.2. Correlated Features

In reality, most data sets contain redundancy and the features are usually correlated. So, we evaluate our models on synthetic data sets with correlated features. We take the similar procedure as in generating the data sets with i.i.d features to first generate one linear-chain CRF model. Then, we use the CRF model to generate 10 data sets of which each sample has 30 relevant features. The 30 relevant features are partitioned into 10 groups. For the features in each group, we first draw a real-value from a standard normal distribution and then 'spoil' the feature with a random Gaussian noise to get 3 correlated features. The noise Gaussian has a zero mean and standard variance 0.05. Here and in all the remaining experiments, we use the sub-gradient method (Ratliff et al., 2007) to solve the QP problem in both M^{3}N and LapM^{3}N. We use the learning rate and complexity constant that are suggested by the authors, that is, $\alpha_t = \frac{1}{2\beta\sqrt{t}}$ and $C = 200\beta$, where β is a parameter we introduced to adjust α_t and C. We do K-fold CV on each data set and take the average over the 10 data sets as the final results. Like (Taskar et al., 2003), in each run we choose one part to do training and test on the rest K-1 parts. We vary K from 20, 10, 7, 5, to 4. In other words, we use 50, 100, about 150, 200, and 250 samples during the training. We use the same grid search to choose λ and β from $\{9, 16, 25, 36, 49, 64\}$ and $\{1, 10, 20, 30, 40, 50, 60\}$ respectively. Results are shown in Figure 3. We can get the same conclusions as in the previous results.

6.2. Real-World OCR Data Set

The OCR data set is partitioned into 10 subsets for 10-fold CV (Taskar et al., 2003; Ratliff et al., 2007). We randomly select N samples from each fold for our experiments. We vary N from 100, 150, 200, to 250, and denote the selected data sets by OCR100, OCR150, OCR200, and OCR250 respectively. When $\beta = 4$ on OCR100 and OCR150, $\beta = 2$ on OCR200 and OCR250, and $\lambda = 36$, results are shown in Figure 4.

Overall, as the number of training data increases, all algorithms achieve lower error rates and smaller variances. Generally, LapM^{3}N consistently outperforms all the other models. M^{3}N outperforms the standard, non-regularized, CRFs and the L_1-CRFs. Again, L_2-CRFs perform comparably to M^{3}N. This is a bit surprising but still reasonable due to the understanding of their only difference on loss functions (Globerson et al., 2007). By examining the prediction accuracy, we can see an obvious over-fitting in CRFs and L_1-CRFs. In contrast, L_2-CRFs are very robust. This is because unlike the synthetic data sets, features in real-world data are usually not completely irrelevant. In this case, putting small weights to zero as in L_1-CRFs will hurt generalization ability and also lead to instability to regularization constants as shown later. Instead, L_2-CRFs do not put small weights to zero but

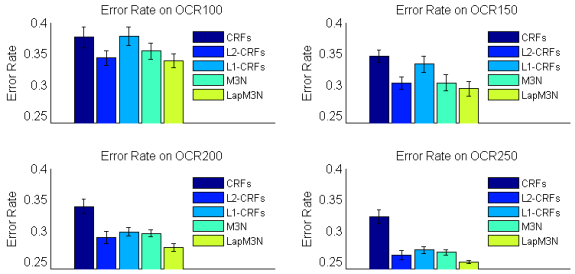

Figure 4. Evaluation results on OCR data set with different numbers of selected data.

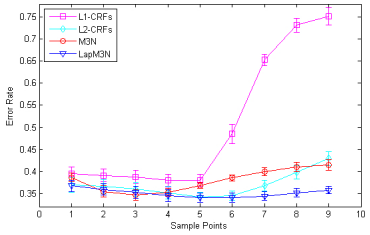

Figure 5. Error rates of different models on OCR100 with different regularization constants. From left to right, the regularization constants are 0.0001, 0.001, 0.01, 0.1, 1, 4, 9, 16, and 25 for L_1-CRFs and L_2-CRFs, and for M^{3}N and LapM^{3}N they are 1, 4, 9, 16, 25, 36, 49, 64, and 81.

shrink them towards zero as in LapM^{3}N. The non-regularized MLE can also easily lead to over-fitting.

6.3. Sensitivity to Regularization Constants

Figure 5 shows the error rates of different models on OCR100. From the results, we can see that the L_1-CRFs are much sensitive to the regularization constants. However, L_2-CRFs, M^{3}N, and LapM^{3}N are much less sensitive. Among all the models, LapM^{3}N is the most stable one. The stability of LapM^{3}N is due to the posterior weighting instead of hard-thresholding to set small weights to zero as in L_1-CRFs.

7. Conclusions

We proposed a *Structured Maximum Entropy Discrimination* formalism for Bayesian max-margin learning in structured prediction. This formalism gives rise to a general class of Bayesian M^{3}Ns and subsumes the standard M^{3}N as a spacial case where the predictive model is assumed to be linear and the parameter prior is a standard normal. We show that the adoption of a Laplace prior of the parameter leads to a Laplace M^{3}N that enjoys properties expected from a sparsified Bayesian M^{3}N. Unlike the L_1-regularized MLE which sets small weights to zeros to achieve sparsity, LapM^{3}N weights the parameters *a posteriori*. Features with smaller weights are shrunk more. This posterior weighting effect makes LapM^{3}N more stable with respect to the magnitudes of the regularization coefficients and more generalizable.

Acknowledgements

We thank Ivor Tsang for inspiring discussions. This work was conceived and completed while J.Z. was a visiting researcher at CMU under a State Scholarship from China, and supports from NSF DBI-0546594 and DBI-0640543 awarded to Eric Xing. J.Z. and B.Z. are also supported by Chinese NSF Grant 60321002; and the National Key Foundation R&D Projects 2004CB318108 and 2007CB311003.

References

Altun, Y., Tsochantaridis, I., & Hofmann, T. (2003). Hidden markov support vector machines. *ICML*.

Andrew, G., & Gao, J. (2007). Scalable training of l_1-regularized log-linear models. *ICML*.

Bartlett, P., Collins, M., Taskar, B., & McAllester, D. (2004). Exponentiated gradient algorithms for large-margin structured classification. *NIPS*.

Bennett, K. P., & Mangasarian, O. L. (1992). Robust linear programming discrimination of two linearly inseparable sets. *Optim. Methods Softw*, 23–34.

Chan, A. B., Vasconcelos, N., & Lanckriet, G. R. G. (2007). Direct convex relaxations of sparse svm. *ICML*.

Dudík, M., Phillips, S. J., & Schapire, R. E. (2007). Maximum entropy density estimation with generalized regularization and an application to species distribution modeling. *JMLR*, 1217–1260.

Figueiredo, M. (2003). Adaptive sparseness for supervised learning. *IEEE Trans. on PAMI*, *25*, 1150–1159.

Globerson, A., Koo, T. Y., Carreras, X., & Collins, M. (2007). Exponentiated gradient algorithms for log-linear structured prediction. *ICML*.

Jaakkola, T., Meila, M., & Jebara, T. (1999). Maximum entropy discrimination. *NIPS*.

Kaban, A. (2007). On bayesian classification with laplace priors. *Pattern Recognition Letters*, *28*, 1271–1282.

Lafferty, J., McCallum, A., & Pereira, F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *ICML*.

Langford, J., Seeger, M., & Megiddo, N. (2001). An improved predictive accuracy bound for averaging classifiers. *ICML*.

Lebanon, G., & Lafferty, J. (2001). Boosting and maximum likelihood for exponential models. *NIPS*.

Lee, S.-I., Ganapathi, V., & Koller, D. (2006). Efficient structure learning of markov networks using l1-regularization. *NIPS*.

Qi, Y. A., Szummer, M., & Minka, T. P. (2005). Bayesian conditional random fields. *AISTATS*.

Ratliff, N. D., Bagnell, J. A., & Zinkevich, M. A. (2007). (online) subgradient methods for structured prediction. *AISTATS*.

Taskar, B., Guestrin, C., & Koller, D. (2003). Max-margin markov networks. *NIPS*.

Taskar, B., Lacoste-Julien, S., & Jordan, M. I. (2006). Structured prediction via the extragradient method. *NIPS*.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *J. Royal. Statist. Soc., B*, 267–288.

Tsochantaridis, I., Hofmann, T., Joachims, T., & Altun, Y. (2004). Support vector machine learning for interdependent and structured output spaces. *ICML*.

Wainwright, M. J., Ravikumar, P., & Lafferty, J. (2006). High-dimensional graphical model selection using l1-regularized logistic regression. *NIPS*.