Journal of Machine Learning Research 10 (2009) 2531-2569 Submitted 3/08; Revised 11/08; Published 11/09

Maximum Entropy Discrimination Markov Networks

Jun Zhu JUNZHU@CS.CMU.EDU
Eric P. Xing EPXING@CS.CMU.EDU
Machine Learning Department
Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh, PA 15213

Editor: Michael Collins

Abstract

The standard maximum margin approach for structured prediction lacks a straightforward probabilistic interpretation of the learning scheme and the prediction rule. Therefore its unique advantages such as dual sparseness and kernel tricks cannot be easily conjoined with the merits of a probabilistic model such as Bayesian regularization, model averaging, and ability to model hidden variables. In this paper, we present a new general framework called *maximum entropy discrimination Markov networks* (MaxEnDNet, or simply, MEDN), which integrates these two approaches and combines and extends their merits. Major innovations of this approach include: 1) It extends the conventional max-entropy discrimination learning of classification rules to a new *structural* max-entropy discrimination paradigm of learning a distribution of Markov networks. 2) It generalizes the extant Markov network structured-prediction rule based on a point estimator of model coefficients to an averaging model akin to a Bayesian predictor that integrates over a learned posterior distribution of model coefficients. 3) It admits flexible entropic regularization of the model during learning. By plugging in different prior distributions of the model coefficients, it subsumes the well-known maximum margin Markov networks (M^3N) as a special case, and leads to a model similar to an L_1-regularized M^3N that is simultaneously primal and dual sparse, or other new types of Markov networks. 4) It applies a modular learning algorithm that combines existing variational inference techniques and convex-optimization based M^3N solvers as subroutines. Essentially, MEDN can be understood as a jointly maximum likelihood and maximum margin estimate of Markov network. It represents the first successful attempt to combine maximum entropy learning (a dual form of maximum likelihood learning) with maximum margin learning of Markov network for structured input/output problems; and the basic principle can be generalized to learning arbitrary graphical models, such as the generative Bayesian networks or models with structured hidden variables. We discuss a number of theoretical properties of this approach, and show that empirically it outperforms a wide array of competing methods for structured input/output learning on both synthetic and real OCR and web data extraction data sets.

Keywords: maximum entropy discrimination, structured input/output model, maximum margin Markov network, graphical models, entropic regularization, L_1 regularization

1. Introduction

Inferring structured predictions based on high-dimensional, often multi-modal and hybrid covariates remains a central problem in data mining (e.g., web-info extraction), machine intelligence (e.g., machine translation), and scientific discovery (e.g., genome annotation). Several recent approaches to this problem are based on learning discriminative graphical models defined on composite features

that explicitly exploit the structured dependencies among input elements and structured interpretational outputs. 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). Various paradigms for training such models based on different loss functions have been explored, including the maximum conditional likelihood learning (Lafferty et al., 2001) and the maximum margin learning (Altun et al., 2003; Taskar et al., 2003; Tsochantaridis et al., 2004), with remarkable success.

The likelihood-based models for structured predictions are usually based on a joint distribution of both input and output variables (Rabiner, 1989) or a conditional distribution of the output given the input (Lafferty et al., 2001). Therefore this paradigm offers a flexible probabilistic framework that can naturally facilitate: hidden variables that capture latent semantics such as a generative hierarchy (Quattoni et al., 2004; Zhu et al., 2008a); Bayesian regularization that imposes desirable biases such as sparseness (Lee et al., 2006; Wainwright et al., 2006; Andrew and Gao, 2007); and Bayesian prediction based on combining predictions across all values of model parameters (i.e., model averaging), which can reduce the risk of overfitting. On the other hand, the margin-based structured prediction models leverage the maximum margin principle and convex optimization formulation underlying the support vector machines, and concentrate directly on the input-output mapping (Taskar et al., 2003; Altun et al., 2003; Tsochantaridis et al., 2004). In principle, this approach can lead to a robust decision boundary due to the dual sparseness (i.e., depending on only a few support vectors) and global optimality of the learned model. However, although arguably a more desirable paradigm for training highly discriminative structured prediction models in a number of application contexts, the lack of a straightforward probabilistic interpretation of the maximum-margin models makes them unable to offer the same flexibilities of likelihood-based models discussed above.

For example, for domains with complex feature space, it is often desirable to pursue a "sparse" representation of the model that leaves out irrelevant features. 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. However, little progress has been made so far on learning sparse MNs or log-linear models in general based on the maximum margin principle. 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 and Mangasarian, 1992) or by adding a cardinality constraint (Chan et al., 2007), generalization of these techniques to structured output space turns out to be non-trivial, as we discuss later in this paper. There is also very little theoretical analysis on the performance guarantee of margin-based models under direct L_1-regularization. Our empirical results as shown in this paper suggest that an L_1-regularized maximum margin Markov network, even when estimable, can be sensitive to the magnitude of the regularization coefficient. Discarding the features that are not completely irrelevant can potentially hurt generalization ability. Another example, it is well known that presence of hidden variables in MNs can cause significant difficulty for maximum margin learning. Indeed, semi-supervised or unsupervised learning of structured maximum margin model remains an open problem of which progress was only made in a few special cases, with usually computationally very expensive algorithms (Xu et al., 2006; Altun et al., 2006; Brefeld and Scheffer, 2006).

In this paper, we propose a general theory of maximum entropy discrimination Markov networks (MaxEnDNet, or simply MEDN) for structured input/output learning and prediction. This formalism offers a formal paradigm for integrating both generative and discriminative principles and

the Bayesian regularization techniques for learning structured prediction models. It integrates the spirit of maximum margin learning from SVM, the design of discriminative structured prediction model in maximum margin Markov networks (M^3N), and the ideas of entropic regularization and model averaging in maximum entropy discrimination methods (Jaakkola et al., 1999). Essentially, MaxEnDNet can be understood as a jointly maximum likelihood and maximum margin estimate of Markov networks. It allows one to learn a *distribution* of structured prediction models that offers a wide range of important advantages over conventional models such as M^3N, including more robust prediction due to an averaging prediction-function based on the learned distribution of models, Bayesian-style regularization that can lead to a model that is simultaneous primal and dual sparse, and allowance of hidden variables and semi-supervised learning based on partially labeled data.

While the formalism of MaxEnDNet is extremely general, our main focus and contributions of this paper will be concentrated on the following results. We will formally define the MaxEnDNet as solving a generalized entropy optimization problem subject to expected margin constraints on the structured predictions, and under an arbitrary prior of feature coefficients; and we derive a general form of the solution to this problem. An interesting insight immediately following this general form is that, a trivial assumption on the prior distribution of the coefficients, that is, a standard normal, reduces the linear MaxEnDNet to the standard M^3N, as shown in Theorem 3. This understanding opens the way to use different priors for MaxEnDNet to achieve more interesting regularization effects. We show that, by using a Laplace prior for the feature coefficients, the resulting Laplace MaxEnDNet (LapMEDN) is effectively an M^3N that is not only dual sparse (i.e., defined by a few support vectors), but also primal sparse (i.e., shrinkage on coefficients corresponding to irrelevant features). We develop a novel variational learning method for the LapMEDN, which leverages on the hierarchical/scale-mixture representation of the Laplace prior (Andrews and Mallows, 1974; Figueiredo, 2003) and the reducibility of MaxEnDNet to M^3N, and combines the variational Bayesian technique with existing convex optimization algorithms developed for M^3N (Taskar et al., 2003; Bartlett et al., 2004; Ratliff et al., 2007). We also provide a formal analysis of the generalization error of the MaxEnDNet, and prove a PAC-Bayes bound on the prediction error by MaxEnDNet. We performed a thorough comparison of the Laplace MaxEnDNet with competing methods, including M^3N (i.e., the Gaussian MaxEnDNet), L_1-regularized M^3N (Zhu et al., 2009b), CRFs, L_1-regularized CRFs, and L_2-regularized CRFs, on both synthetic and real structured input/output data. The Laplace MaxEnDNet exhibits mostly superior, and sometimes comparable performance in all scenarios been tested.

As demonstrated in our recent work (Zhu et al., 2008c, 2009a), MaxEnDNet is not limited to fully observable MNs, but can readily facilitate jointly maximum entropy and maximum margin learning of partially observed structured I/O models, and directed graphical models such as the supervised latent Dirichlet allocation (LDA). Due to space limit, we leave these instantiations and generalizations to future papers.

The rest of the paper is structured as follows. In the next section, we review the basic structured prediction formalism and set the stage for our model. Section 3 presents the general theory of maximum entropy discrimination Markov networks and some basic theoretical results, followed by two instantiations of the general MaxEnDNet, the Gaussian MaxEnDNet and the Laplace MaxEnDNet. Section 4 offers a detailed discussion of the primal and dual sparsity property of Laplace MaxEnDNet. Section 5 presents a novel iterative learning algorithm based on variational approximation and convex optimization. In Section 6, we briefly discuss the generalization bound of MaxEnDNet.

Then, we show empirical results on both synthetic and real OCR and web data extraction data sets in Section 7. Section 8 discusses some related work and Section 9 concludes this paper.

2. Preliminaries

In structured prediction problems such as natural language parsing, image annotation, or DNA decoding, one aims to learn a function $h : \mathcal{X} \to \mathcal{Y}$ that maps 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, unlike a standard classification problem, $\mathbf{y}$ is a multivariate prediction consisting of multiple labeling elements. Let L denote the cardinality of the output, and m_l where $l = 1, \ldots, L$ denote the arity of each element, then $\mathcal{Y} = \mathcal{Y}_1 \times \cdots \times \mathcal{Y}_L$ with $\mathcal{Y}_l = \{a_1, \ldots, a_{m_l}\}$ represents a combinatorial space of structured interpretations of the multi-facet objects in the inputs. For example, $\mathcal{Y}$ could correspond to the space of all possible instantiations of the parse trees 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 $y_l \in \mathcal{Y}_l$ within $\mathbf{y}$ must be determined in the context of other labels $y_{l' \neq l}$, rather than independently as in classification, in order to arrive at a globally satisfactory and consistent prediction.

Let $F : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ represent a discriminant function over the input-output pairs from which one can define the predictive function, and let $\mathcal{H}$ denote the space of all possible F. A common choice of F is a linear model, $F(\mathbf{x}, \mathbf{y}; \mathbf{w}) = g(\mathbf{w}^\top \mathbf{f}(\mathbf{x}, \mathbf{y}))$, where $\mathbf{f} = [f_1 \ldots f_K]^\top$ is a K-dimensional column vector of the feature functions $f_k : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$, and $\mathbf{w} = [w_1 \ldots w_K]^\top$ is the corresponding vector of the weights of the feature functions. Typically, a structured prediction model chooses an optimal estimate $\mathbf{w}^\star$ by minimizing some loss function $J(\mathbf{w})$, and defines a predictive function in terms of an optimization problem that maximizes $F(\,\cdot\,; \mathbf{w}^\star)$ over the response variable $\mathbf{y}$ given an input $\mathbf{x}$:

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

where $\mathcal{Y}(\mathbf{x}) \subseteq \mathcal{Y}$ is the feasible subset of structured labels for the input $\mathbf{x}$. Here, we assume that $\mathcal{Y}(\mathbf{x})$ is finite for any $\mathbf{x}$.

Depending on the specific choice of $g(\cdot)$ (e.g., linear, or log linear), and the loss function $J(\mathbf{w})$ (e.g., likelihood, or margin-based loss) for estimating the parameter $\mathbf{w}^\star$, incarnations of the general structured prediction formalism described above can be seen in classical generative models such as the HMM (Rabiner, 1989) where $g(\cdot)$ can be an exponential family distribution function and $J(\mathbf{w})$ is the (negative) joint likelihood of the input and its labeling; and in recent discriminative models such as CRFs (Lafferty et al., 2001), where $g(\cdot)$ is a Boltzmann machine and $J(\mathbf{w})$ is the (negative) conditional likelihood of the structured labeling given input; and the M^{3}N (Taskar et al., 2003), where $g(\cdot)$ is an identity function and $J(\mathbf{w})$ is a loss defined on the margin between the true labeling and any other feasible labeling in $\mathcal{Y}(\mathbf{x})$. Our approach toward a more general discriminative training is based on a maximum entropy principle that allows an elegant combination of the discriminative maximum margin learning with the generative Bayesian regularization and hierarchical modeling, and we consider the more general problem of finding a distribution of $F(\,\cdot\,; \mathbf{w})$ over $\mathcal{H}$ that enables a convex combination of discriminant functions for robust structured prediction.

Before delving into the exposition of the proposed approach, we end this section with a brief recapitulation of the basic M^{3}N, upon which the proposed approach is built. Under a max-margin

framework, given a set of fully observed 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):

$$\begin{aligned} \text{P0 } (\text{M}^3\text{N}): \quad & \min_{\mathbf{w},\xi} \ \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N} \xi_i \\ \text{s.t. } \forall i, \forall \mathbf{y} \neq \mathbf{y}^i: \quad & \mathbf{w}^\top \Delta\mathbf{f}_i(\mathbf{y}) \geq \Delta\ell_i(\mathbf{y}) - \xi_i, \ \xi_i \geq 0 \ , \end{aligned}$$

where $\Delta\mathbf{f}_i(\mathbf{y}) = \mathbf{f}(\mathbf{x}^i, \mathbf{y}^i) - \mathbf{f}(\mathbf{x}^i, \mathbf{y})$ and $\Delta F_i(\mathbf{y}; \mathbf{w}) = \mathbf{w}^\top \Delta\mathbf{f}_i(\mathbf{y})$ is the "margin" between the true label $\mathbf{y}^i$ and a prediction $\mathbf{y}$, $\Delta\ell_i(\mathbf{y})$ is a labeling loss with respect to $\mathbf{y}^i$, and ξ_i represents a slack variable that absorbs errors in the training data. Various forms of the labeling loss have been proposed in the literature (Tsochantaridis et al., 2004). In this paper, we adopt the *hamming loss* used by Taskar et al. (2003): $\Delta\ell_i(\mathbf{y}) = \sum_{j=1}^{L} \mathbb{I}(y_j \neq y_j^i)$, where $\mathbb{I}(\cdot)$ is an indicator function that equals to one if the argument is true and zero otherwise. The problem P0 is not directly solvable by using a standard constrained optimization toolbox because the feasible space for $\mathbf{w}$,

$$\mathcal{F}_0 = \left\{ \mathbf{w} : \mathbf{w}^\top \Delta\mathbf{f}_i(\mathbf{y}) \geq \Delta\ell_i(\mathbf{y}) - \xi_i; \ \forall i, \forall \mathbf{y} \neq \mathbf{y}^i \right\},$$

is defined by $O(N|\mathcal{Y}|)$ number of constraints, and $\mathcal{Y}$ is exponential to the size of the input $\mathbf{x}$. Exploring sparse dependencies among individual labels y_l in $\mathbf{y}$, as reflected in the specific design of the feature functions (e.g., based on pair-wise labeling potentials in a pair-wise Markov network), and the convex duality of the objective, efficient optimization 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 to P0. As described shortly, these algorithms can be directly employed as subroutines in solving our proposed model.

3. Maximum Entropy Discrimination Markov Networks

Instead of learning a point estimator of $\mathbf{w}$ as in M^{3}N, in this paper, we take a Bayesian-style approach and learn a distribution $p(\mathbf{w})$, in a max-margin manner. For prediction, we employ a convex combination of all possible models $F(\,\cdot\,; \mathbf{w}) \in \mathcal{H}$ based on $p(\mathbf{w})$, that is:

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

Now, the open question underlying this averaging prediction rule is how we can devise an appropriate loss function and constraints over $p(\mathbf{w})$, in a similar spirit as the margin-based scheme over $\mathbf{w}$ in P0, that lead to an optimum estimate of $p(\mathbf{w})$. In the sequel, we present *Maximum Entropy Discrimination Markov Networks*, a novel framework that facilitates the estimation of a Bayesian-style regularized *distribution* of M^{3}Ns defined by $p(\mathbf{w})$. As we show below, this new Bayesian-style max-margin learning formalism offers several advantages such as simultaneous primal and dual sparsity, PAC-Bayesian generalization guarantee, and estimation robustness. Note that the MaxEnDNet is different from the traditional Bayesian methods for discriminative structured prediction such as the Bayesian CRFs (Qi et al., 2005), where the likelihood function is well defined. Here, our approach is of a "Bayesian-style" because it learns and uses a "posterior" distribution of all predictive models instead of choosing one model according to some criterion, but the learning algorithm is not

based on the Bayes theorem, but a maximum entropy principle that biases towards a distribution that makes less additional assumptions over a given prior over the predictive models. We emphasize that this "posterior" is different from, and should not be confused with, the conventional Bayesian posterior defined according to the Bayes rule.

It is well-known that exponential family distributions can be expressed variationally as the solution to a maximum entropy estimation subject to moment constraints, and the maximum entropy estimation of parameters can be understood as a dual to the maximum likelihood estimation of the parameters of exponential family distributions. Thus our combination of the maximum entropy principle with the maximum margin principle to be presented in the sequel offers an elegant way of achieving jointly maximum margin and maximum likelihood effects on learning structured input/output Markov networks, and in fact, general exponential family graphical models.

3.1 Structured Maximum Entropy Discrimination

Given a training set $\mathcal{D}$ of structured input-output pairs, analogous to the feasible space $\mathcal{F}_0$ for the weight vector $\mathbf{w}$ in a standard M^{3}N (c.f., problem P0), we define the feasible subspace $\mathcal{F}_1$ for the weight distribution $p(\mathbf{w})$ by a set of *expected* margin constraints:

$$\mathcal{F}_1 = \Big\{ p(\mathbf{w}) : \int p(\mathbf{w})[\Delta F_i(\mathbf{y};\mathbf{w}) - \Delta\ell_i(\mathbf{y})]\,\mathrm{d}\mathbf{w} \geq -\xi_i,\ \forall i, \forall \mathbf{y} \neq \mathbf{y}^i \Big\}.$$

We learn the optimum $p(\mathbf{w})$ from $\mathcal{F}_1$ based on a *structured maximum entropy discrimination principle* generalized from the maximum entropy discrimination (Jaakkola et al., 1999). Under this principle, the optimum $p(\mathbf{w})$ corresponds to the distribution that minimizes its relative entropy with respect to some chosen prior p_0, as measured by the Kullback-Leibler divergence between p and p_0: $KL(p||p_0) = \langle \log(p/p_0) \rangle_p$, where $\langle\cdot\rangle_p$ denotes the expectations with respect to p. If p_0 is uniform, then minimizing this KL-divergence is equivalent to maximizing the entropy $H(p) = -\langle \log p \rangle_p$. A natural information theoretic interpretation of this formulation is that we favor a distribution over the discriminant function class $\mathcal{H}$ that bears minimum assumptions among all feasible distributions in $\mathcal{F}_1$. The p_0 is a regularizer that introduces an appropriate bias, if necessary.

To accommodate non-separable cases in the discriminative prediction problem, instead of minimizing the usual KL, we optimize the *generalized entropy* (Dudík et al., 2007; Lebanon and 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 term can be understood as an additional "potential" in the maximum entropy principle. Putting everything together, we can now state a general formalism based on the following maximum entropy discrimination Markov network framework:

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

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

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. In addition, the expectations $\langle F(\mathbf{x},\mathbf{y};\mathbf{w})\rangle_{p(\mathbf{w})}$ are required to be bounded in order for F to be a meaningful model. Thus, the problem P1 satisfies the *Slater's condition*[1] (Boyd and Vandenberghe, 2004, chap. 5), which together with the convexity make P1 enjoy nice properties, such as strong duality and the existence of solutions. The problem P1 can be solved via applying the calculus of variations to the Lagrangian to obtain a variational extremum, followed by a dual transformation of P1. We state the main results below as a theorem, followed by a brief proof that lends many insights into the solution to P1 which we will explore in subsequent analysis.

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

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

where $Z(\alpha)$ *is a normalization factor and the Lagrange multipliers* $\alpha_i(\mathbf{y})$ *(corresponding to the constraints in* $\mathcal{F}_1$*) can be obtained by solving the dual problem of P1:*

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

where $U^\star(\cdot)$ *is the conjugate of the slack function* $U(\cdot)$*, that is,* $U^\star(\alpha) = \sup_\xi (\sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y})\xi_i - U(\xi))$.

Proof (*sketch*) Since the problem P1 is a convex program and satisfies the Slater's condition, we can form a Lagrangian function, whose saddle point gives the optimal solution of P1 and D1, by introducing a non-negative dual variable $\alpha_i(\mathbf{y})$ for each constraint in $\mathcal{F}_1$ and another non-negative dual variable c for the normalization constraint $\int p(\mathbf{w})\,\mathrm{d}\mathbf{w} = 1$. Details are deferred to Appendix B.1. ■

Since the problem P1 is a convex program and satisfies the Slater's condition, the saddle point of the Lagrangian function is the KKT point of P1. From the KKT conditions (Boyd and Vandenberghe, 2004, Chap. 5), it can be shown that the above solution enjoys *dual sparsity*, that is, only a few Lagrange multipliers will be non-zero, which correspond to the active constraints whose equality holds, analogous to the support vectors in SVM. Thus MaxEnDNet enjoys a similar generalization property as the M^{3}N and SVM due to the the small "effective size" of the margin constraints. But it is important to realize that this does not mean that the learned model is "primal-sparse", that is, only a few elements in the weight vector $\mathbf{w}$ are non-zero. We will return to this point in Section 4.

For a closed proper convex function $\phi(\mu)$, its conjugate is defined as $\phi^\star(\nu) = \sup_\mu[\nu^\top\mu - \phi(\mu)]$. In the problem D1, by convex duality (Boyd and Vandenberghe, 2004), the log normalizer $\log Z(\alpha)$ can be shown to be the conjugate of the KL-divergence. If the slack function is $U(\xi) = C\|\xi\| =$

1. Since $\langle F(\mathbf{x},\mathbf{y};\mathbf{w})\rangle_{p(\mathbf{w})}$ are bounded and $\xi_i \geq 0$, there always exists a ξ, which is large enough to make the pair $(p(\mathbf{w}),\xi)$ satisfy the Slater's condition.

$C\sum_i \xi_i$, 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 separable. Ignoring this inequality does not affect the solution since the special case $\xi = 0$ is still included. Thus, the Lagrange 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). In this case, expectations with respect to $p(\xi)$ are taken on both sides of all the constraints in $\mathcal{F}_1$. Take the duality, and the dual function of U is another log normalizer. More details were provided by Jaakkola et al. (1999). Some other U functions and their dual functions are studied by Lebanon and Lafferty (2001) and Dudík et al. (2007).

Unlike most extant structured discriminative models including the highly successful M^{3}N, which rely on a point estimator of the parameters, the MaxEnDNet model derived above gives an optimum parameter distribution, which is used to make prediction via the rule (2). Indeed, as we will show shortly, the MaxEnDNet is strictly more general than the M^{3}N and subsumes the later as a special case. But more importantly, the MaxEnDNet in its full generality offers a number of important advantages while retaining all the merits of the M^{3}N. **First**, MaxEnDNet admits a prior that can be designed to introduce useful regularization effects, such as a primal sparsity bias. **Second**, the MaxEnDNet prediction is based on model averaging and therefore enjoys a desirable smoothing effect, with a uniform convergence bound on generalization error. **Third**, MaxEnDNet offers a principled way to incorporate *hidden* generative models underlying the structured predictions, but allows the predictive model to be discriminatively trained based on partially labeled data. In the sequel, we analyze the first two points in detail; exploration of the third point is beyond the scope of this paper, and can be found in Zhu et al. (2008c), where a *partially observed* MaxEnDNet (PoMEN) is developed, which combines (possibly latent) generative model and discriminative training for structured prediction.

3.2 Gaussian MaxEnDNet

As Equation (3) suggests, different choices of the parameter prior can lead to different MaxEnDNet models for predictive parameter distribution. In this subsection and the following one, we explore a few common choices, e.g., Gaussian and Laplace priors.

We first show that, when the parameter prior is set to be a standard normal, MaxEnDNet leads to a predictor that is identical to that of the M^{3}N. This somewhat surprising reduction offers an important insight for understanding the property of MaxEnDNet. Indeed this result should not be totally unexpected given the striking isomorphisms of the opt-problem P1, the feasible space $\mathcal{F}_1$, and the predictive function h_1 underlying a MaxEnDNet, to their counterparts P0, $\mathcal{F}_0$, and h_0, respectively, underlying an M^{3}N. The following theorem makes our claim explicit.

Theorem 3 (Gaussian MaxEnDNet: Reduction of MEDN to M^{3}N) *Assuming* $F(\mathbf{x},\mathbf{y};\mathbf{w}) = \mathbf{w}^\top \mathbf{f}(\mathbf{x},\mathbf{y})$, $U(\xi) = C\sum_i \xi_i$, *and* $p_0(\mathbf{w}) = \mathcal{N}(\mathbf{w}|0,I)$, *where I denotes an identity matrix, then the posterior distribution is* $p(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mu,I)$, *where* $\mu = \sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y})\Delta\mathbf{f}_i(\mathbf{y})$, *and the Lagrange multipliers* $\alpha_i(\mathbf{y})$ *in* $p(\mathbf{w})$ *are obtained by solving the following dual problem, which is isomorphic to the dual form of the M^{3}N:*

$$\max_{\alpha} \sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y})\Delta\ell_i(\mathbf{y}) - \frac{1}{2}\| \sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y})\Delta\mathbf{f}_i(\mathbf{y})\|^2$$

$$\text{s.t.} \quad \sum_{\mathbf{y} \neq \mathbf{y}^i} \alpha_i(\mathbf{y}) = C;\ \alpha_i(\mathbf{y}) \geq 0,\ \forall i,\ \forall \mathbf{y} \neq \mathbf{y}^i,$$

where $\Delta \mathbf{f}_i(\mathbf{y}) = \mathbf{f}(\mathbf{x}^i, \mathbf{y}^i) - \mathbf{f}(\mathbf{x}^i, \mathbf{y})$ *as in P0. When applied to* h_1*,* $p(\mathbf{w})$ *leads to a predictive function that is identical to* $h_0(\mathbf{x}; \mathbf{w})$ *given by Equation (1).*

Proof See Appendix B.2 for details. ■

The above theorem is stated in the duality form. We can also show the following equivalence in the primal form.

Corollary 4 *Under the same assumptions as in Theorem 3, the mean* μ *of the posterior distribution* $p(\mathbf{w})$ *under a Gaussian MaxEnDNet is obtained by solving the following primal problem:*

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

Proof See Appendix B.3 for details. ■

Theorem 3 and Corollary 4 both show that in the supervised learning setting, the M^{3}N is a special case of MaxEnDNet when the slack function is linear and the parameter prior is a standard normal. As we shall see later, this connection renders many existing techniques for solving the M^{3}N directly applicable for solving the MaxEnDNet.

3.3 Laplace MaxEnDNet

Recent trends in pursuing "sparse" graphical models has led to the emergence of regularized version of CRFs (Andrew and 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 fully explored or evaluated in the context of maximum margin learning, although some existing methods can be extended to achieve sparse max-margin estimators, as explained below.

One possible way to learn a sparse M^{3}N is to adopt the strategy of L_1-SVM (Bennett and Mangasarian, 1992; Zhu et al., 2004) and directly use an L_1 instead of the L_2-norm of $\mathbf{w}$ in the loss function (see appendix A for a detailed description of this formulation and the duality derivation). However, the primal problem of an L_1-regularized M^{3}N is not directly solvable using a standard optimization toolbox by re-formulating it as an LP problem due to the exponential number of constraints; solving the dual problem, which now has only a polynomial number of constraints as in the dual of M^{3}N, is also non-trivial due to the complicated form of the constraints. The constraint generation methods (Tsochantaridis et al., 2004) are possible. However, although such methods have been shown to be efficient for solving the QP problem in the standard M^{3}N, our preliminary empirical results show that such a scheme with an LP solver for the L_1-regularized M^{3}N can be extremely expensive for a non-trivial real data set. Another type of possible solvers are based on a projection to L_1-ball (Duchi et al., 2008), such as the gradient descent (Ratliff et al., 2007) and the dual extragradient (Taskar et al., 2006) methods.

The MaxEnDNet interpretation of the M^{3}N offers an alternative strategy that resembles Bayesian regularization (Tipping, 2001; Kaban, 2007) in maximum likelihood estimation, where shrinkage effects can be introduced by appropriate priors over the model parameters. As Theorem 3 reveals, an M^{3}N corresponds to a Gaussian MaxEnDNet that admits a standard normal prior for the weight vector $\mathbf{w}$. According to the standard Bayesian regularization theory, to achieve a sparse estimate of a model, in the posterior distribution of the feature weights, the weights of irrelevant features should peak around zero with very small variances. However, the isotropy of the variances in all dimensions of the feature space under a standard normal prior makes it infeasible for the resulting M^{3}N to adjust the variances in different dimensions to fit a sparse model. Alternatively, now we employ a Laplace prior for $\mathbf{w}$ to learn a Laplace MaxEnDNet. We show in the sequel that, the parameter posterior $p(\mathbf{w})$ under a Laplace MaxEnDNet has a shrinkage effect on small weights, which is similar to directly applying an L_1-regularizer on an M^{3}N. Although exact learning of a Laplace MaxEnDNet is also intractable, we show that this model can be efficiently approximated by a variational inference procedure based on existing methods.

The Laplace prior of $\mathbf{w}$ is expressed as $p_0(\mathbf{w}) = \prod_{k=1}^{K} \frac{\sqrt{\lambda}}{2} e^{-\sqrt{\lambda}|w_k|} = (\frac{\sqrt{\lambda}}{2})^K e^{-\sqrt{\lambda}\|\mathbf{w}\|}$. This density function is heavy tailed and peaked at zero; thus, it encodes a prior belief that the distribution of $\mathbf{w}$ is strongly peaked around zero. Another nice property of the Laplace density is that it is log-concave, or the negative logarithm is convex, which can be exploited to obtain a convex estimation problem analogous to LASSO (Tibshirani, 1996).

Theorem 5 (Laplace MaxEnDNet: a sparse M^{3}N) *Assuming* $F(\mathbf{x}, \mathbf{y}; \mathbf{w}) = \mathbf{w}^\top \mathbf{f}(\mathbf{x}, \mathbf{y})$, $U(\xi) = C\sum_i \xi_i$, *and* $p_0(\mathbf{w}) = \prod_{k=1}^{K} \frac{\sqrt{\lambda}}{2} e^{-\sqrt{\lambda}|w_k|} = (\frac{\sqrt{\lambda}}{2})^K e^{-\sqrt{\lambda}\|\mathbf{w}\|}$, *then the Lagrange multipliers* $\alpha_i(\mathbf{y})$ *in* $p(\mathbf{w})$ *(as defined in Theorem 2) are obtained by solving the following dual problem:*

$$\max_{\alpha} \sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y}) \Delta\ell_i(\mathbf{y}) - \sum_{k=1}^{K} \log \frac{\lambda}{\lambda - \eta_k^2}$$
$$\text{s.t.} \sum_{\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y}) = C;\ \alpha_i(\mathbf{y}) \geq 0,\ \forall i,\ \forall \mathbf{y} \neq \mathbf{y}^i.$$

where $\eta_k = \sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y}) \Delta\mathbf{f}_i^k(\mathbf{y})$, *and* $\Delta\mathbf{f}_i^k(\mathbf{y}) = f_k(\mathbf{x}^i, \mathbf{y}^i) - f_k(\mathbf{x}^i, \mathbf{y})$ *represents the kth component of* $\Delta\mathbf{f}_i(\mathbf{y})$. *Furthermore, constraints* $\eta_k^2 < \lambda,\ \forall k$, *must be satisfied.*

Since several intermediate results from the proof of this Theorem will be used in subsequent presentations, we provide the complete proof below. Our proof is based on a hierarchical representation of the Laplace prior. As noted by Andrews and Mallows (1974), the Laplace distribution $p(w) = \frac{\sqrt{\lambda}}{2} e^{-\sqrt{\lambda}|w|}$ is equivalent to a two-layer hierarchical Gaussian-exponential model, where w follows a zero-mean Gaussian distribution $p(w|\tau) = \mathcal{N}(w|0, \tau)$ and the variance τ admits an exponential hyper-prior density,

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

This alternative form straightforwardly leads to the following new representation of our multivariate Laplace prior for the parameter vector $\mathbf{w}$ in MaxEnDNet:

$$p_0(\mathbf{w}) = \prod_{k=1}^{K} p_0(w_k) = \prod_{k=1}^{K} \int p(w_k|\tau_k) p(\tau_k|\lambda)\, \mathrm{d}\tau_k = \int p(\mathbf{w}|\tau) p(\tau|\lambda)\, \mathrm{d}\tau, \tag{4}$$

where $p(\mathbf{w}|\tau) = \prod_{k=1}^{K} p(w_k|\tau_k)$ and $p(\tau|\lambda) = \prod_{k=1}^{K} p(\tau_k|\lambda)$ represent multivariate Gaussian and exponential, respectively, and $d\tau \triangleq d\tau_1 \cdots d\tau_K$.

Proof (*of Theorem 5*) Substitute the hierarchical representation of the Laplace prior (Equation 4) into $p(\mathbf{w})$ in Theorem 2, and we get the normalization factor $Z(\alpha)$ as follows,

$$\begin{aligned}
Z(\alpha) &= \int\int p(\mathbf{w}|\tau)p(\tau|\lambda)\,d\tau \cdot \exp\{\mathbf{w}^\top\eta - \sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y})\Delta\ell_i(\mathbf{y})\}\,d\mathbf{w} \\
&= \int p(\tau|\lambda)\int p(\mathbf{w}|\tau)\cdot\exp\{\mathbf{w}^\top\eta - \sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y})\Delta\ell_i(\mathbf{y})\}\,d\mathbf{w}\,d\tau \\
&= \int p(\tau|\lambda)\int \mathcal{N}(\mathbf{w}|0,A)\exp\{\mathbf{w}^\top\eta - \sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y})\Delta\ell_i(\mathbf{y})\}\,d\mathbf{w}\,d\tau \\
&= \int p(\tau|\lambda)\exp\{\frac{1}{2}\eta^\top A\eta - \sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y})\Delta\ell_i(\mathbf{y})\}d\tau \\
&= \exp\{-\sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y})\Delta\ell_i(\mathbf{y})\}\prod_{k=1}^{K}\int\frac{\lambda}{2}\exp(-\frac{\lambda}{2}\tau_k)\exp(\frac{1}{2}\eta_k^2\tau_k)d\tau_k \\
&= \exp\{-\sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y})\Delta\ell_i(\mathbf{y})\}\prod_{k=1}^{K}\frac{\lambda}{\lambda-\eta_k^2}, \qquad (5)
\end{aligned}$$

where $A = \mathrm{diag}(\tau_k)$ is a diagonal matrix and η is a column vector with η_k defined as in Theorem 5. The last equality is due to the moment generating function of an exponential distribution. The constraint $\eta_k^2 < \lambda,\ \forall k$ is needed in this derivation to avoid the integration going infinity. Substituting the normalization factor derived above into the general dual problem D1 in Theorem 2, and using the same argument of the convex conjugate of $U(\xi) = C\sum_i \xi_i$ as in Theorem 3, we arrive at the dual problem in Theorem 5. ■

It can be shown that the dual objective function of Laplace MaxEnDNet in Theorem 5 is concave.[2] But since each η_k depends on all the dual variables α and η_k^2 appears within a logarithm, the optimization problem underlying Laplace MaxEnDNet would be very difficult to solve. The SMO (Taskar et al., 2003) and the exponentiated gradient methods (Bartlett et al., 2004) developed for the QP dual problem of M^{3}N cannot be easily applied here. Thus, we will turn to a variational approximation method, as shown in Section 5. For completeness, we end this section with a corollary similar to the Corollary 4, which states the primal optimization problem underlying the MaxEnDNet with a Laplace prior. As we shall see, the primal optimization problem in this case is complicated and provides another perspective of the hardness of solving the Laplace MaxEnDNet.

Corollary 6 *Under the same assumptions as in Theorem 5, the mean μ of the posterior distribution $p(\mathbf{w})$ under a Laplace MaxEnDNet is obtained by solving the following primal problem:*

$$\begin{aligned}
&\min_{\mu,\xi}\ \sqrt{\lambda}\sum_{k=1}^{K}\Big(\sqrt{\mu_k^2+\frac{1}{\lambda}} - \frac{1}{\sqrt{\lambda}}\log\frac{\sqrt{\lambda\mu_k^2+1}+1}{2}\Big) + C\sum_{i=1}^{N}\xi_i \\
&\text{s.t.}\ \ \mu^\top\Delta\mathbf{f}_i(\mathbf{y}) \geq \Delta\ell_i(\mathbf{y}) - \xi_i;\ \xi_i \geq 0,\ \ \forall i,\ \forall\mathbf{y}\neq\mathbf{y}^i.
\end{aligned}$$

2. η_k^2 is convex over α because it is the composition of $f(x) = x^2$ with an affine mapping. So, $\lambda - \eta_k^2$ is concave and $\log(\lambda - \eta_k^2)$ is also concave due to the composition rule (Boyd and Vandenberghe, 2004).

Proof The proof requires the result of Corollary 7. We defer it to Appendix B.4. ■

Since the "norm"[3]

$$\sum_{k=1}^{K}\Big(\sqrt{\mu_k^2+\frac{1}{\lambda}}-\frac{1}{\sqrt{\lambda}}\log\frac{\sqrt{\lambda\mu_k^2+1}+1}{2}\Big)\triangleq\|\mu\|_{KL}$$

corresponds to the KL-divergence between $p(\mathbf{w})$ and $p_0(\mathbf{w})$ under a Laplace MaxEnDNet, we will refer to it as a *KL-norm* and denote it by $\|\cdot\|_{KL}$ in the sequel. This KL-norm is different from the L_2-norm as used in M^{3}N, but is closely related to the L_1-norm, which encourages a sparse estimator. In the following section, we provide a detailed analysis of the sparsity of Laplace MaxEnDNet resulted from the regularization effect from this norm.

4. Entropic Regularization and Sparse M^{3}N

Comparing to the structured prediction law h_0 due to an M^{3}N, which enjoys dual sparsity (i.e., few support vectors), the h_1 defined by a Laplace MaxEnDNet is not only dual-sparse, but also primal sparse; that is, features that are insignificant will experience strong shrinkage on their corresponding weight w_k.

The primal sparsity of h_1 achieved by the Laplace MaxEnDNet is due to a shrinkage effect resulting from the *Laplacian entropic regularization*. In this section, we take a close look at this regularization effect, in comparison with other common regularizers, such as the L_2-norm in M^{3}N (which is equivalent to the Gaussian MaxEnDNet), and the L_1-norm that at least in principle could be directly applied to M^{3}N. Since our main interest here is the sparsity of the structured prediction law h_1, we examine the posterior mean under $p(\mathbf{w})$ via exact integration. It can be shown that under a Laplace MaxEnDNet, $p(\mathbf{w})$ exhibits the following posterior shrinkage effect.

Corollary 7 (Entropic Shrinkage) *The posterior mean of the Laplace MaxEnDNet has the following form:*

$$\langle w_k\rangle_p=\frac{2\eta_k}{\lambda-\eta_k^2},\ \forall 1\le k\le K, \tag{6}$$

where $\eta_k=\sum_{i,\mathbf{y}\neq\mathbf{y}^i}\alpha_i(\mathbf{y})(f_k(\mathbf{x}^i,\mathbf{y}^i)-f_k(\mathbf{x}^i,\mathbf{y}))$ *and* $\eta_k^2<\lambda,\ \forall k$.

Proof Using the integration result in Equation (5), we can get:

$$\frac{\partial\log Z}{\partial\alpha_i(\mathbf{y})}=v^\top\Delta\mathbf{f}_i(\mathbf{y})-\Delta\ell_i(\mathbf{y}), \tag{7}$$

where v is a column vector and $v_k=\frac{2\eta_k}{\lambda-\eta_k^2},\ \forall 1\le k\le K$. An alternative way to compute the derivatives is using the definition of Z: $Z=\int p_0(\mathbf{w})\cdot\exp\{\mathbf{w}^\top\eta-\sum_{i,\mathbf{y}\neq\mathbf{y}^i}\alpha_i(\mathbf{y})\Delta\ell_i(\mathbf{y})\}\,\mathrm{d}\mathbf{w}$. We can get:

$$\frac{\partial\log Z}{\partial\alpha_i(\mathbf{y})}=\langle\mathbf{w}\rangle_p^\top\Delta\mathbf{f}_i(\mathbf{y})-\Delta\ell_i(\mathbf{y}). \tag{8}$$

3. This is not exactly a norm because the positive scalability does not hold. But the KL-norm is non-negative due to the non-negativity of KL-divergence. In fact, by using the inequality $e^x\ge 1+x$, we can show that each component $(\sqrt{\mu_k^2+\frac{1}{\lambda}}-\frac{1}{\sqrt{\lambda}}\log\frac{\sqrt{\lambda\mu_k^2+1}+1}{2})$ is monotonically increasing with respect to μ_k^2 and $\|\mu\|_{KL}\ge K/\sqrt{\lambda}$, where the equality holds only when $\mu=0$. Thus, $\|\mu\|_{KL}$ penalizes large weights. For convenient comparison with the popular L_2 and L_1 norms, we call it a KL-norm.

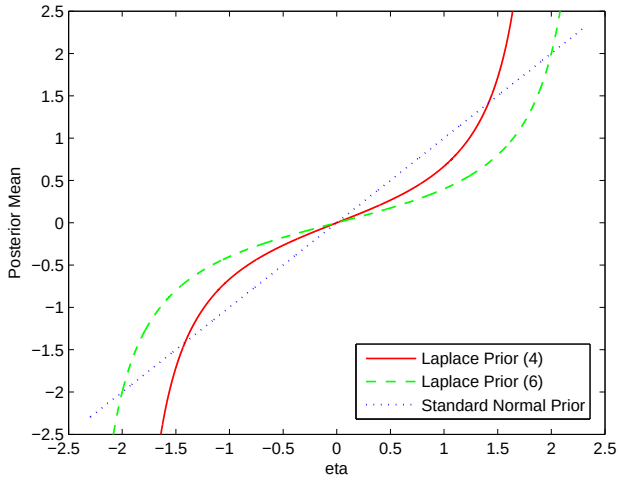


Figure 1: Posterior means with different priors against their corresponding $\eta = \sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y})\Delta\mathbf{f}_i(\mathbf{y})$. Note that the η for different priors are generally different because of the different dual parameters.

Comparing Equations (7) and (8), we get $\langle\mathbf{w}\rangle_p = v$, that is, $\langle w_k\rangle_p = \frac{2\eta_k}{\lambda-\eta_k^2}$, $\forall 1 \leq k \leq K$. The constraints $\eta_k^2 < \lambda$, $\forall k$ are required to get a finite normalization factor as shown in Equation (5). ■

Here η is isomorphic to an unregularized estimate of the feature weight vector which directly comes from a linear combination of support vectors (and therefore not sparsified). A plot of the relationship between $\langle w_k\rangle_p$ under a Laplace MaxEnDNet and the corresponding η_k revealed by Corollary 7 is shown in Figure 1 (for example, the red curve), from which we can see that, the smaller the η_k is, the more shrinkage toward zero is imposed on $\langle w_k\rangle_p$.

This entropic shrinkage effect on $\mathbf{w}$ is not present in the standard M^{3}N, and the Gaussian MaxEnDNet. Recall that by definition, the vector $\eta \triangleq \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})\Delta\mathbf{f}_i(\mathbf{y})$ is determined by the dual parameters $\alpha_i(\mathbf{y})$ obtained by solving a model-specific dual problem. When the $\alpha_i(\mathbf{y})$'s are obtained by solving the dual of the standard M^{3}N, it can be shown that the optimum point solution of the parameters $\mathbf{w}^\star = \eta$. When the $\alpha_i(\mathbf{y})$'s are obtained from the dual of the Gaussian MaxEnDNet, Theorem 3 shows that the posterior mean of the parameters $\langle\mathbf{w}\rangle_{p_{\text{Gaussian}}} = \eta$. (As we have already pointed out, since these two dual problems are isomorphic, the $\alpha_i(\mathbf{y})$'s for M^{3}N and Gaussian MaxEnDNet are identical, hence the resulting η's are the same.) In both cases, there is no shrinkage along any particular dimension of the parameter vector $\mathbf{w}$ or of the mean vector of $p(\mathbf{w})$. Therefore, although both M^{3}N and Gaussian MaxEnDNet enjoy the dual sparsity, because the KKT conditions imply that most of the dual parameters $\alpha_i(\mathbf{y})$'s are zero, $\mathbf{w}^\star$ and $\langle\mathbf{w}\rangle_{p_{\text{Gaussian}}}$ are not primal sparse. From Equation (6), we can conclude that the Laplace MaxEnDNet is also dual sparse, because its mean $\langle\mathbf{w}\rangle_{p_{\text{Laplace}}}$ can be uniquely determined by η. But the shrinkage effect on different components of the $\langle\mathbf{w}\rangle_{p_{\text{Laplace}}}$ vector causes $\langle\mathbf{w}\rangle_{p_{\text{Laplace}}}$ to be also primal sparse.

A comparison of the posterior mean estimates of $\mathbf{w}$ under MaxEnDNet with three different priors versus their associated η is shown in Figure 1. The three priors in question are, a standard normal, a Laplace with $\lambda = 4$, and a Laplace with $\lambda = 6$. It can be seen that, under the entropic

regularization with a Laplace prior, the $\langle \mathbf{w} \rangle_p$ gets shrunk toward zero when η is small. The larger the λ value is, the greater the shrinkage effect. For a fixed λ, the shape of the shrinkage curve (i.e., the $\langle \mathbf{w} \rangle_p - \eta$ curve) is smoothly nonlinear, but no component is explicitly discarded, that is, no weight is set explicitly to zero. In contrast, for the Gaussian MaxEnDNet, which is equivalent to the standard M^{3}N, there is no such a shrinkage effect.

Corollary 6 offers another perspective of how the Laplace MaxEnDNet relates to the L_1-norm M^{3}N, which yields a sparse estimator. Note that as λ goes to infinity, the KL-norm $\|\mu\|_{KL}$ approaches $\|\mu\|_1$, that is, the L_1-norm.[4] This means that the MaxEnDNet with a Laplace prior will be (nearly) the same as the L_1-M^{3}N if the regularization constant λ is large enough.

A more explicit illustration of the entropic regularization under a Laplace MaxEnDNet, comparing to the conventional L_1 and L_2 regularization over an M^{3}N, can be seen in Figure 2, where the feasible regions due to the three different norms used in the regularizer are plotted in a two dimensional space. Specifically, it shows (1) L_2-norm: $w_1^2 + w_2^2 \leq 1$; (2) L_1-norm: $|w_1| + |w_2| \leq 1$; and (3) KL-norm:[5] $\sqrt{w_1^2 + 1/\lambda} + \sqrt{w_2^2 + 1/\lambda} - (1/\sqrt{\lambda}) \log(\sqrt{\lambda w_1^2 + 1}/2 + 1/2) - (1/\sqrt{\lambda}) \log(\sqrt{\lambda w_1^2 + 1}/2 + 1/2) \leq b$, where b is a parameter to make the boundary pass the $(0,1)$ point for easy comparison with the L_2 and L_1 curves. It is easy to show that b equals to $\sqrt{1/\lambda} + \sqrt{1 + 1/\lambda} - (1/\sqrt{\lambda}) \log(\sqrt{\lambda + 1}/2 + 1/2)$. It can be seen that the L_1-norm boundary has sharp turning points when it passes the axises, whereas the L_2 and KL-norm boundaries turn smoothly at those points. This is the intuitive explanation of why the L_1-norm directly gives sparse estimators, whereas the L_2-norm and KL-norm due to a Laplace prior do not. But as shown in Figure 2(b), when the λ gets larger and larger, the KL-norm boundary moves closer and closer to the L_1-norm boundary. When $\lambda \to \infty$, $\sqrt{w_1^2 + 1/\lambda} + \sqrt{w_2^2 + 1/\lambda} - (1/\sqrt{\lambda}) \log(\sqrt{\lambda w_1^2 + 1}/2 + 1/2) - (1/\sqrt{\lambda}) \log(\sqrt{\lambda w_1^2 + 1}/2 + 1/2) \to |w_1| + |w_2|$ and $b \to 1$, which yields exactly the L_1-norm in the two dimensional space. Thus, under the linear model assumption of the discriminant functions $F(\cdot\,; \mathbf{w})$, our framework can be seen as a smooth relaxation of the L_1-M^{3}N.

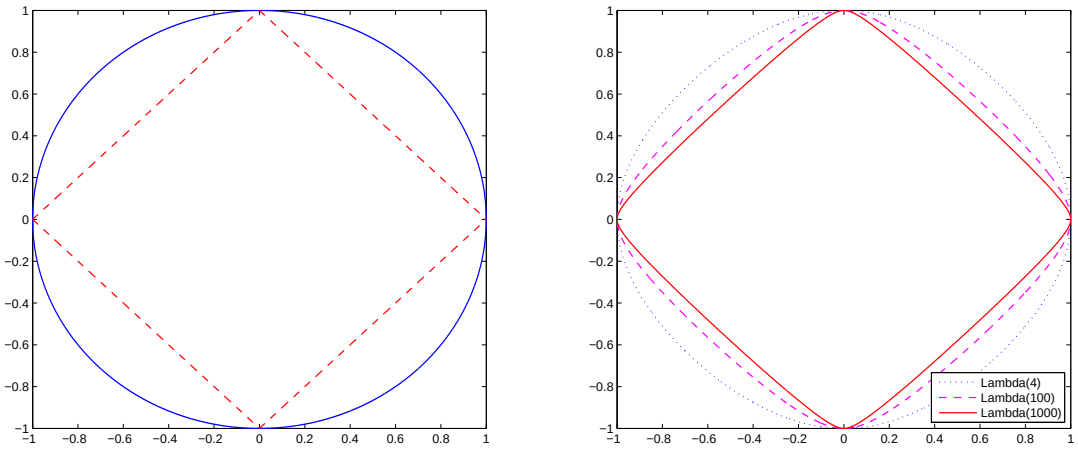
5. Variational Learning of Laplace MaxEnDNet

Although Theorem 2 seems to offer a general closed-form solution to $p(\mathbf{w})$ under an arbitrary prior $p_0(\mathbf{w})$, in practice the Lagrange multipliers $\alpha_i(\mathbf{y})$ in $p(\mathbf{w})$ can be very hard to estimate from the dual problem D1 except for a few special choices of $p_0(\mathbf{w})$, such as a normal as shown in Theorem 3, which can be easily generalized to any normal prior. When $p_0(\mathbf{w})$ is a Laplace prior, as we have shown in Theorem 5 and Corollary 6, the corresponding dual problem or primal problem involves a complex objective function that is difficult to optimize. Here, we present a variational method for an approximate learning of the Laplace MaxEnDNet.

Our approach is built on the hierarchical interpretation of the Laplace prior as shown in Equation (4). Replacing the $p_0(\mathbf{w})$ in Problem P1 with Equation (4), and applying the Jensen's inequality, we get an upper bound of the KL-divergence:

$$KL(p||p_0) \qquad = -H(p) - \langle \log \int p(\mathbf{w}|\tau) p(\tau|\lambda)\, \mathrm{d}\tau \rangle_p$$

4. As $\lambda \to \infty$, the logarithm terms in $\|\mu\|_{KL}$ disappear because of the fact that $\frac{\log x}{x} \to 0$ when $x \to \infty$.
5. The curves are drawn with a symbolic computational package to solve an equation of the form: $2x - \log x = a$, where x is the variable to be solved and a is a constant.



(a) (b)

Figure 2: (a) L_2-norm (solid line) and L_1-norm (dashed line); (b) KL-norm with different Laplace priors.

$$\leq -H(p) - \langle \int q(\tau) \log \frac{p(\mathbf{w}|\tau)p(\tau|\lambda)}{q(\tau)} \, \mathrm{d}\tau \rangle_p$$
$$\triangleq \mathcal{L}(p(\mathbf{w}), q(\tau)),$$

where $q(\tau)$ is a variational distribution used to approximate $p(\tau|\lambda)$. The upper bound is in fact a KL-divergence: $\mathcal{L}(p(\mathbf{w}), q(\tau)) = KL(p(\mathbf{w})q(\tau)||p(\mathbf{w}|\tau)p(\tau|\lambda))$. Thus, $\mathcal{L}$ is convex over $p(\mathbf{w})$, and $q(\tau)$, respectively, but not necessarily joint convex over $(p(\mathbf{w}), q(\tau))$.

Substituting this upper bound for the KL-divergence in P1, we now solve the following Variational MaxEnDNet problem,

$$\text{P1}' \text{ (vMEDN)}: \qquad \min_{p(\mathbf{w}) \in \mathcal{F}_1; q(\tau); \xi} \mathcal{L}(p(\mathbf{w}), q(\tau)) + U(\xi).$$

P1$'$ can be solved with an iterative minimization algorithm alternating between optimizing over $(p(\mathbf{w}), \xi)$ and $q(\tau)$, as outlined in Algorithm 1, and detailed below.

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

$$\begin{aligned} p(\mathbf{w}) \quad &\propto \exp\{\int q(\tau) \log p(\mathbf{w}|\tau) \, \mathrm{d}\tau - b\} \cdot \exp\{\mathbf{w}^\top \eta - \sum_{i, \mathbf{y} \neq \mathbf{y}^i} \alpha_i(\mathbf{y}) \Delta \ell_i(\mathbf{y})\} \\ &\propto \exp\{-\frac{1}{2} \mathbf{w}^\top \langle A^{-1} \rangle_q \mathbf{w} - b + \mathbf{w}^\top \eta - \sum_{i, \mathbf{y} \neq \mathbf{y}^i} \alpha_i(\mathbf{y}) \Delta \ell_i(\mathbf{y})\} \\ &= \mathcal{N}(\mathbf{w}|\mu, \Sigma), \end{aligned}$$

Algorithm 1 Variational MaxEnDNet

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 $\langle \mathbf{w} \rangle_p^1 \leftarrow 0$, $\Sigma^1 \leftarrow I$
for $t = 1$ **to** $T-1$ **do**
  Step 1: solve (9) or (10) for $\langle \mathbf{w} \rangle_p^{t+1} = \Sigma^t \eta$; update $\langle \mathbf{w}\mathbf{w}^\top \rangle_p^{t+1} \leftarrow \Sigma^t + \langle \mathbf{w} \rangle_p^{t+1} (\langle \mathbf{w} \rangle_p^{t+1})^\top$.
  Step 2: use (11) to update $\Sigma^{t+1} \leftarrow \text{diag}(\sqrt{\frac{\langle w_k^2 \rangle_p^{t+1}}{\lambda}})$.
end for

where $\eta = \sum_{i,\mathbf{y} \neq \mathbf{y}^i} \alpha_i(\mathbf{y}) \Delta \mathbf{f}_i(\mathbf{y})$, $A = \text{diag}(\tau_k)$, and $b = KL(q(\tau)||p(\tau|\lambda))$ is a constant. The posterior mean and variance are $\langle \mathbf{w} \rangle_p = \mu = \Sigma\eta$ and $\Sigma = (\langle A^{-1} \rangle_q)^{-1} = \langle \mathbf{w}\mathbf{w}^\top \rangle_p - \langle \mathbf{w} \rangle_p \langle \mathbf{w} \rangle_p^\top$, respectively. Note that this posterior distribution is also a normal distribution. Analogous to the proof of Theorem 3, we can derive that the dual parameters α are estimated by solving the following dual problem:

$$\begin{aligned} &\max_{\alpha} \sum_{i,\mathbf{y} \neq \mathbf{y}^i} \alpha_i(\mathbf{y}) \Delta \ell_i(\mathbf{y}) - \frac{1}{2} \eta^\top \Sigma \eta \\ &\text{s.t.} \sum_{\mathbf{y} \neq \mathbf{y}^i} \alpha_i(\mathbf{y}) = C;\ \alpha_i(\mathbf{y}) \geq 0,\ \forall i,\ \forall \mathbf{y} \neq \mathbf{y}^i. \end{aligned} \tag{9}$$

This dual problem is now a standard quadratic program symbolically identical to the dual of an M^3N, and can be directly solved using existing algorithms developed for M^3N, such as the SMO (Taskar et al., 2003) and the exponentiated gradient (Bartlett et al., 2004) methods. Alternatively, we can solve the following primal problem:

$$\begin{aligned} &\min_{\mathbf{w},\xi} \frac{1}{2} \mathbf{w}^\top \Sigma^{-1} \mathbf{w} + C \sum_{i=1}^N \xi_i \\ &\text{s.t. } \mathbf{w}^\top \Delta \mathbf{f}_i(\mathbf{y}) \geq \Delta \ell_i(\mathbf{y}) - \xi_i;\ \xi_i \geq 0,\ \ \forall i,\ \forall \mathbf{y} \neq \mathbf{y}^i. \end{aligned} \tag{10}$$

Based on the proof of Corollary 4, it is easy to show that the solution of the problem (10) leads to the posterior mean of $\mathbf{w}$ under $p(\mathbf{w})$, which will be used to do prediction by h_1. The primal problem can be solved with the subgradient (Ratliff et al., 2007), cutting-plane (Tsochantaridis et al., 2004), or extragradient (Taskar et al., 2006) method.

Step 2: Keep $p(\mathbf{w})$ fixed, optimize P1′ with respect to $q(\tau)$. Taking the derivative of $\mathcal{L}$ with respect to $q(\tau)$ and set it to zero, we get:

$$q(\tau) \quad \propto p(\tau|\lambda) \exp\{\langle \log p(\mathbf{w}|\tau) \rangle_p\}.$$

Since both $p(\mathbf{w}|\tau)$ and $p(\tau|\lambda)$ can be written as a product of univariate Gaussian and univariate exponential distributions, respectively, over each dimension, $q(\tau)$ also factorizes over each dimension: $q(\tau) = \prod_{k=1}^K q(\tau_k)$, where each $q(\tau_k)$ can be expressed as:

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

The same distribution has been derived by Kaban (2007), and similar to the hierarchical representation of a Laplace distribution we can get the normalization factor: $\int \mathcal{N}(\sqrt{\langle w_k^2\rangle_p}|0,\tau_k)\cdot \frac{\lambda}{2}\exp(-\frac{1}{2}\lambda\tau_k)\,\mathrm{d}\tau_k = \frac{\sqrt{\lambda}}{2}\exp(-\sqrt{\lambda\langle w_k^2\rangle_p})$. Also, we can calculate the expectations $\langle\tau_k^{-1}\rangle_q$ which are required in calculating $\langle A^{-1}\rangle_q$ as follows,

$$\langle\frac{1}{\tau_k}\rangle_q = \int \frac{1}{\tau_k} q(\tau_k)\,\mathrm{d}\tau_k = \sqrt{\frac{\lambda}{\langle w_k^2\rangle_p}}. \tag{11}$$

We iterate between the above two steps until convergence. Due to the convexity (not joint convexity) of the upper bound, the algorithm is guaranteed to converge to a local optimum. Then, we apply the posterior distribution $p(\mathbf{w})$, which is in the form of a normal distribution, to make prediction using the averaging prediction law in Equation (2). Due to the shrinkage effect of the Laplacian entropic regularization discussed in Section 4, for irrelevant features, the variances should converge to zeros and thus lead to a sparse estimation of $\mathbf{w}$. To summarize, 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 the standard M^{3}N; then, in the second step we update the covariance matrix in the QP problem with an exponential hyper-prior on the variance.

6. Generalization Bound

The PAC-Bayes theory for averaging classifiers (McAllester, 1999; Langford et al., 2001) provides a theoretical motivation to learn an averaging model for classification. In this section, we extend the classic PAC-Bayes theory on binary classifiers to MaxEnDNet, and analyze the generalization performance of the structured prediction rule h_1 in Equation (2). In order to prove an error bound for h_1, the following mild assumption on the boundedness of discriminant function $F(\,\cdot\,;\mathbf{w})$ is necessary, that is, there exists a positive constant c, such that,

$$\forall \mathbf{w},\quad F(\,\cdot\,;\mathbf{w}) \in \mathcal{H}:\quad \mathcal{X}\times\mathcal{Y} \to [-c,c].$$

Recall that the averaging structured prediction function under the MaxEnDNet is defined as $h(\mathbf{x},\mathbf{y}) = \langle F(\mathbf{x},\mathbf{y};\mathbf{w})\rangle_{p(\mathbf{w})}$. Let's define the predictive margin of an instance $(\mathbf{x},\mathbf{y})$ under 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, h makes a wrong prediction on $(\mathbf{x},\mathbf{y})$ only if $M(h,\mathbf{x},\mathbf{y}) \leq 0$. Let Q denote a distribution over $\mathcal{X}\times\mathcal{Y}$, and let $\mathcal{D}$ represent a sample of N instances randomly drawn from Q. With these definitions, we have the following structured version of PAC-Bayes theorem.

Theorem 8 (PAC-Bayes Bound of MaxEnDNet) *Let p_0 be any continuous probability distribution over $\mathcal{H}$ and let $\delta\in(0,1)$. If $F(\,\cdot\,;\mathbf{w})\in\mathcal{H}$ is bounded by $\pm c$ as above, then with probability at least $1-\delta$, for a random sample $\mathcal{D}$ of N instances from Q, for every distribution p over $\mathcal{H}$, and for all margin thresholds $\gamma>0$:*

$$\Pr_Q(M(h,\mathbf{x},\mathbf{y})\leq 0) \leq \Pr_{\mathcal{D}}(M(h,\mathbf{x},\mathbf{y})\leq\gamma) + O\Big(\sqrt{\frac{\gamma^{-2}KL(p||p_0)\ln(N|\mathcal{Y}|)+\ln N+\ln\delta^{-1}}{N}}\Big),$$

where $\Pr_Q(.)$ *and* $\Pr_{\mathcal{D}}(.)$ *represent the probabilities of events over the true distribution Q, and over the empirical distribution of* $\mathcal{D}$*, respectively.*

The proof of Theorem 8 follows the same spirit of the proof of the original PAC-Bayes bound, but with a number of technical extensions dealing with structured outputs and margins. See appendix B.5 for the details.

Recently, McAllester (2007) presents a *stochastic* max-margin structured prediction model, which is different from the averaging predictor under the MaxEnDNet model, by defining/designing a "posterior" distribution from which a model is sampled to make prediction, and achieves a PAC-Bayes bound which holds for arbitrary models sampled from the particular posterior distribution. Langford and Shawe-Taylor (2003) show an interesting connection between the PAC-Bayes bounds for averaging classifiers and stochastic classifiers, again by designing a posterior distribution. But our posterior distribution is solved with MaxEnDNet and is generally different from those designed by McAllester (2007) and Langford and Shawe-Taylor (2003).

7. Experiments

In this section, we present empirical evaluations of the proposed Laplace MaxEnDNet (LapMEDN) on both synthetic and real data sets. We compare LapMEDN with M^{3}N (i.e., the Gaussian MaxEnDNet), L_1-regularized M^{3}N (L_1-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 (Liu and Nocedal, 1989) and its variant (Andrew and Gao, 2007) to solve the optimization problem of CRFs, L_1-CRFs, and L_2-CRFs. For M^{3}N and LapMEDN, we use the exponentiated gradient method (Bartlett et al., 2004) to solve the dual QP problem; and we also use the sub-gradient method (Ratliff et al., 2007) to solve the corresponding primal problem. To the best of our knowledge, no formal description, implementation, and evaluation of the L_1-M^{3}N exist in the literature, therefore for comparison purpose we had to develop this model and algorithm anew. Details of our work along this line was reported in Zhu et al. (2009b), which is beyond the scope of this paper. But briefly, for our experiments on synthetic data, we implemented the constraint generating method (Tsochantaridis et al., 2004) which uses MOSEK to solve an equivalent LP re-formulation of L_1-M^{3}N. However, this approach is extremely slow on larger problems; therefore on real data we instead applied the sub-gradient method (Ratliff et al., 2007) with a projection to an L_1-ball (Duchi et al., 2008) to solve the larger L_1-M^{3}N based on the equivalent re-formulation with an L_1-norm constraint (i.e., the second formulation in Appendix A).

7.1 Evaluation on Synthetic Data

We first evaluate all the competing models on synthetic data where the true structured predictions are known. Here, we consider sequence data, that is, each input $\mathbf{x}$ is a sequence $(x_1, \ldots, x_L)$, and each component x_l is a d-dimensional vector of input features. The synthetic data are generated from pre-specified conditional random field models with either i.i.d. instantiations of the input features (i.e., elements in the d-dimensional feature vectors) or correlated (i.e., structured) instantiations of the input features, from which samples of the structured output $\mathbf{y}$, that is, a sequence $(y_1, \ldots, y_L)$, can be drawn from the conditional distribution $p(\mathbf{y}|\mathbf{x})$ defined by the CRF based on a Gibbs sampler.

7.1.1 I.I.D. INPUT FEATURES

The first experiment is conducted on synthetic sequence data with 100 i.i.d. input features (i.e., $d = 100$). We generate three types of data sets with 10, 30, and 50 relevant input features, respectively. For each type, we randomly generate 10 linear-chain CRFs with 8 binary labeling states (i.e., $L = 8$ and $\mathcal{Y}_l = \{0,1\}$). The feature functions include: a real valued state-feature function over a one dimensional input feature and a class label; and 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 input features from a standard normal distribution, and then apply a Gibbs sampler (based on the conditional distribution of the generated CRFs) to assign a labeling sequence with 5000 iterations.

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

The results are shown in Figure 3. All the results of the LapMEDN are achieved with 3 iterations of the variational learning algorithm. From the results, we can see that under different settings LapMEDN consistently outperforms M^{3}N and performs comparably with L_1-CRFs and L_1-M^{3}N, both of which encourage a sparse estimate; and both the L_1-CRFs and L_2-CRFs outperform the un-regularized CRFs, especially in the cases where the number of training data is small. One interesting result is that the 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) that the L_2-regularized maximum likelihood estimation of CRFs has a similar convex dual as that of the M^{3}N, and the only difference is the loss they try to optimize, that is, CRFs optimize the log-loss while M^{3}N optimizes the hinge-loss. Another interesting observation is that when there are very few relevant features, L_1-M^{3}N performs the best (slightly better than LapMEDN); but as the number of relevant features increases LapMEDN performs slightly better than the L_1-M^{3}N. Finally, as the number of training data increases, all the algorithms consistently achieve better performance.

7.1.2 CORRELATED INPUT FEATURES

In reality, most data sets contain redundancies and the input features are usually correlated. So, we evaluate our models on synthetic data sets with correlated input features. We take the similar procedure as in generating the data sets with i.i.d. features to first generate 10 linear-chain CRF models. Then, each CRF is used to generate a data set that contain 1000 instances, each with 100 input features of which 30 are relevant to the output. The 30 relevant input 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 corrupt 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 the variational learning algorithm of LapMEDN. 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

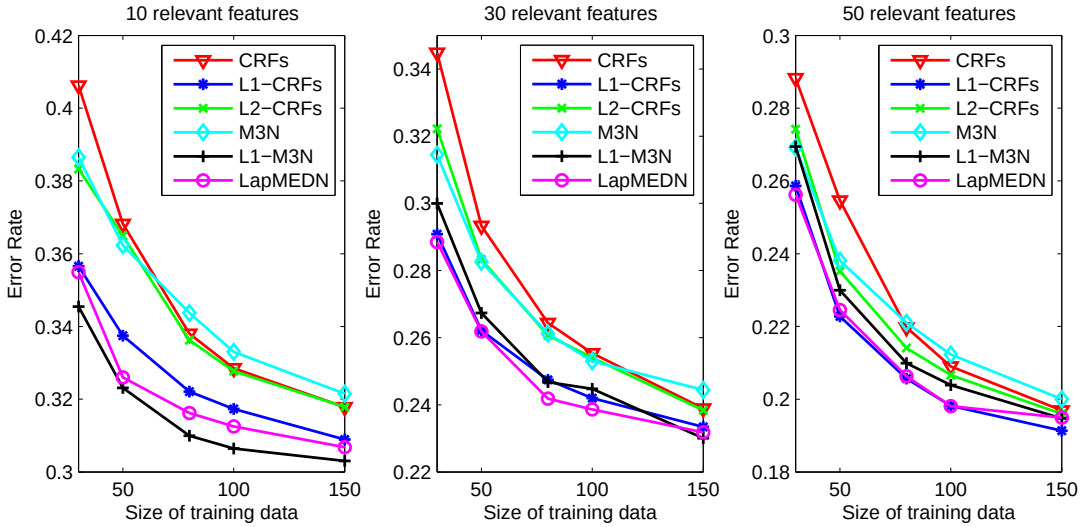


Figure 3: Evaluation results on data sets with i.i.d features.

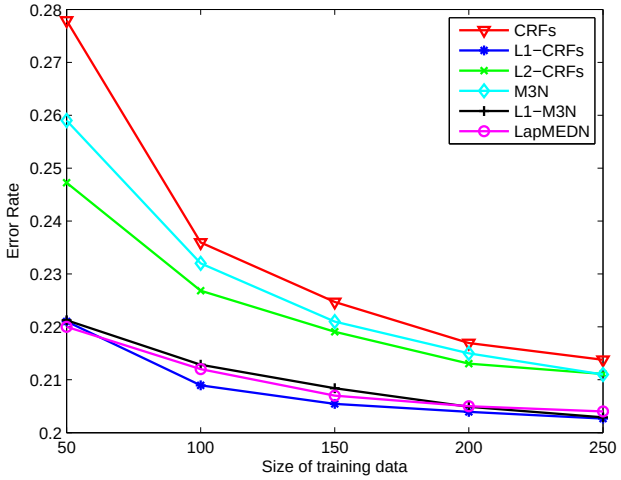


Figure 4: Results on data sets with 30 relevant features.

10 data sets as the final results. Like the method of 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 4. We can get the same conclusions as in the previous results.

Figure 5 shows the true weights of the corresponding 200 state feature functions in the model that generates the first data set, and the average of estimated weights of these features under all competing models fitted from the first data set. All the averages are taken over 10 fold cross-validation. From the plots (2 to 7) of the average model weights, we can see that: for the last 140 state feature functions, which correspond to the last 70 irrelevant features, their average weights under LapMEDN (averaged posterior means $\mathbf{w}$ in this case), L_1-M^{3}N and L_1-CRFs are extremely small, while CRFs and L_2-CRFs can have larger values; for the first 60 state feature functions, which correspond to the 30 relevant features, the overall weight estimation under LapMEDN is similar to

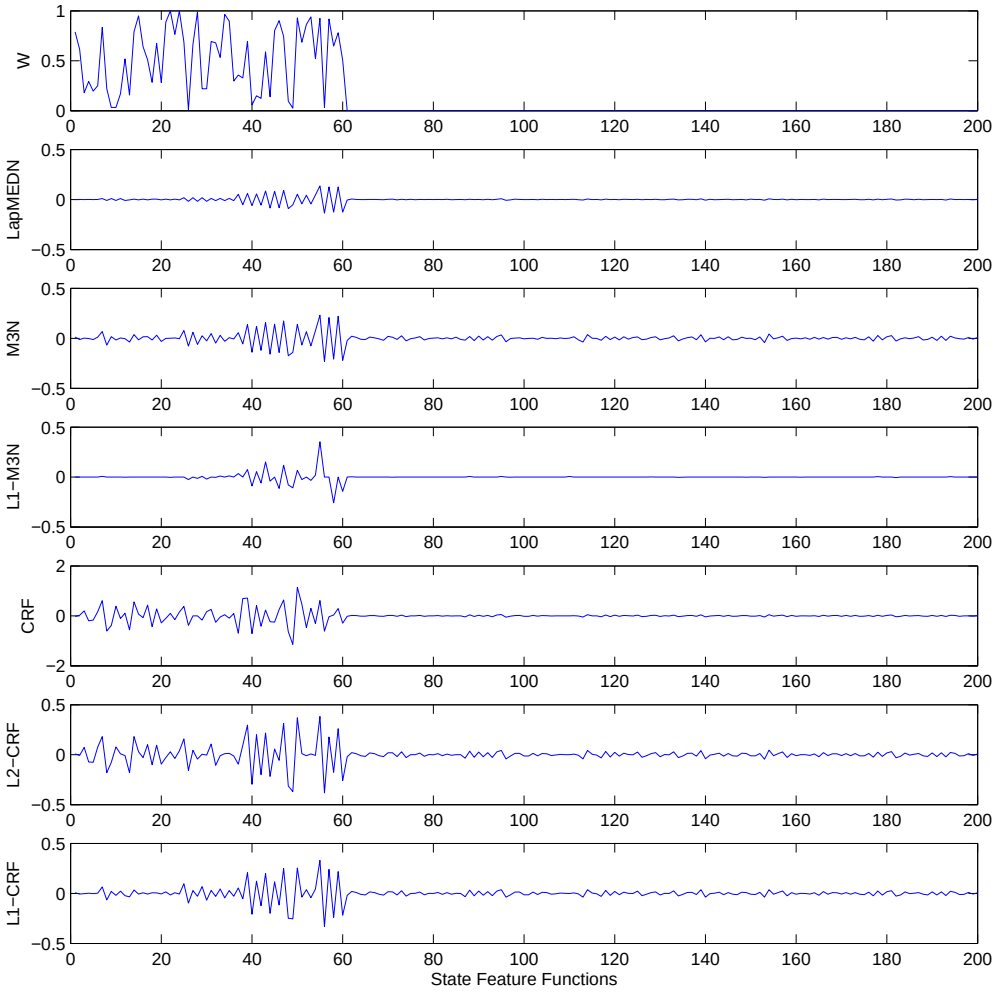


Figure 5: From top to bottom, plot 1 shows the weights of the state feature functions in the linear-chain CRF model from which the data are generated; plot 2 to plot 7 show the average weights of the learned LapMEDN, M^{3}N, L_1-M^{3}N, CRFs, L_2-CRFs, and L_1-CRFs over 10 fold CV, respectively.

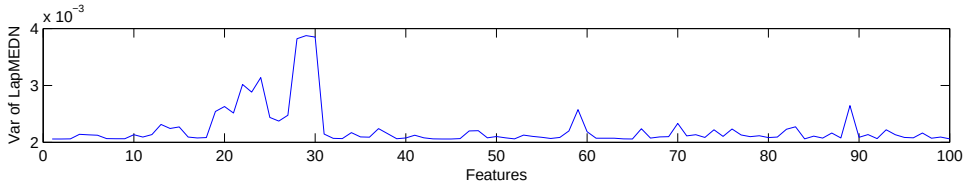


Figure 6: The average variances of the features on the first data set by LapMEDN.

that of the sparse L_1-CRFs and L_1-M^{3}N, but appear to exhibit more shrinkage. Noticeably, CRFs and L_2-CRFs both have more feature functions with large average weights. Note that all the models have quite different average weights from the model (see the first plot) that generates the data. This is because we use a stochastic procedure (i.e., Gibbs sampler) to assign labels to the generated data samples instead of using the labels that are predicted by the model that generates the data. In fact, if we use the model that generates the data to do prediction on its generated data, the error rate is about 0.5. Thus, the learned models, which get lower error rates, are different from the model that generates the data. Figure 6 shows the variances of the 100-dimensional input features (since the variances of the two feature functions that correspond to the same input feature are the same, we collapse each pair into one point) learned by LapMEDN. Again, the variances are the averages over 10 fold cross-validation. From the plot, we can see that the LapMEDN can recover the correlation among the features to some extend, e.g., for the first 30 correlated features, which are the relevant to the output, the features in the same group tend to have similar (average) variances in LapMEDN, whereas there is no such correlation among all the other features. From these observations in both Figure 5 and 6, we can conclude that LapMEDN can reasonably recover the sparse structures in the input data.

7.2 Real-World OCR Data Set

The OCR data set is partitioned into 10 subsets for 10-fold CV as in Taskar et al. (2003) and Ratliff et al. (2007). We randomly select N samples from each fold and put them together to do 10-fold CV. We vary N from 100, 150, 200, to 250, and denote the selected data sets by OCR100, OCR150, OCR200, and OCR250, respectively. On these data sets and the web data as in Section 7.4, our implementation of the cutting plane method for L_1-M^{3}N is extremely slow. The warm-start simplex method of MOSEK does not help either. For example, if we stop the algorithm with 600 iterations on OCR100, then it will take about 20 hours to finish the 10 fold CV. Even with more than 5 thousands of constraints in each training, the performance is still very bad (the error rate is about 0.45). Thus, we turn to an approximate projected sub-gradient method to solve the L_1-M^{3}N by combining the on-line subgradient method (Ratliff et al., 2007) and the efficient L_1-ball projection algorithm (Duchi et al., 2008). The projected sub-gradient method does not work so well as the cutting plane method on the synthetic data sets. That's why we use two different methods.

For $\beta = 4$ on OCR100 and OCR150, $\beta = 2$ on OCR200 and OCR250, and $\lambda = 36$, the results are shown in Figure 7. We can see that as the number of training instances increases, all the algorithms get lower error rates and smaller variances. Generally, the LapMEDN 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 with M^{3}N. This is a bit surprising but still reasonable due to the understanding of their only difference on the loss functions (Globerson et al., 2007) as we have stated. By examining the prediction accuracy during the learning, we can see an obvious over-fitting in CRFs and L_1-CRFs as shown in Figure 8. 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 shrink them towards zero as in the LapMEDN. The non-regularized maximum likelihood estimation can easily lead to over-fitting too. For the two sparse models, the results suggest the

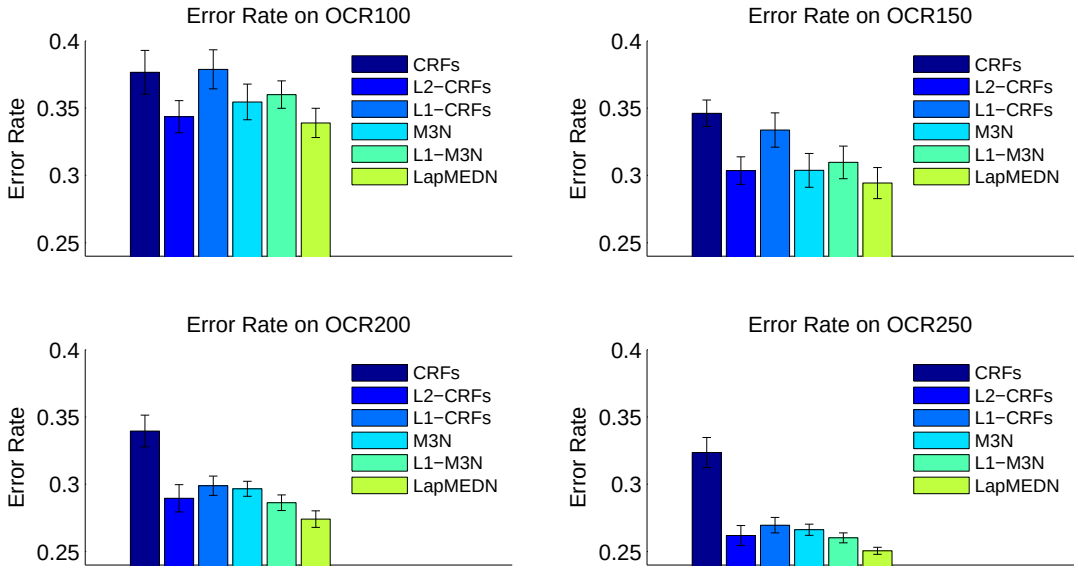


Figure 7: Evaluation results on OCR data set with different numbers of selected data.

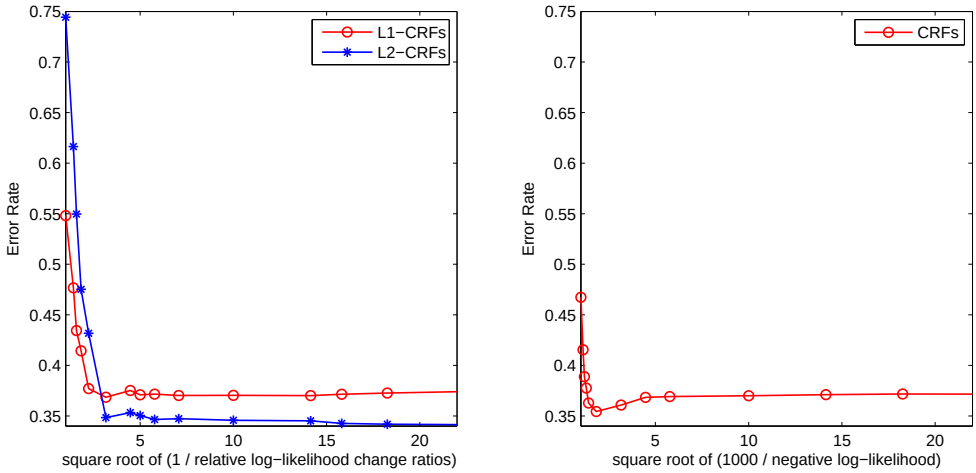


Figure 8: The error rates of CRF models on test data during the learning. For the left plot, the horizontal axis is $\sqrt{1/ratioLL}$, where *ratioLL* is the relative change ratios of the log-likelihood and from left to right, the change ratios are 1, 0.5, 0.4, 0.3, 0.2, 0.1, 0.05, 0.04, 0.03, 0.02, 0.01, 0.005, 0.004, 0.003, 0.002, 0.001, 0.0005, 0.0004, 0.0003, 0.0002, 0.0001, and 0.00005; for the right plot, the horizontal axis is $\sqrt{1000/negLL}$, where *negLL* is the negative log-likelihood, and from left to right *negLL* are 1000, 800, 700, 600, 500, 300, 100, 50, 30, 10, 5, 3, 1, 0.5, 0.3, 0.1, 0.05, 0.03, 0.01, 0.005, 0.003, and 0.002.

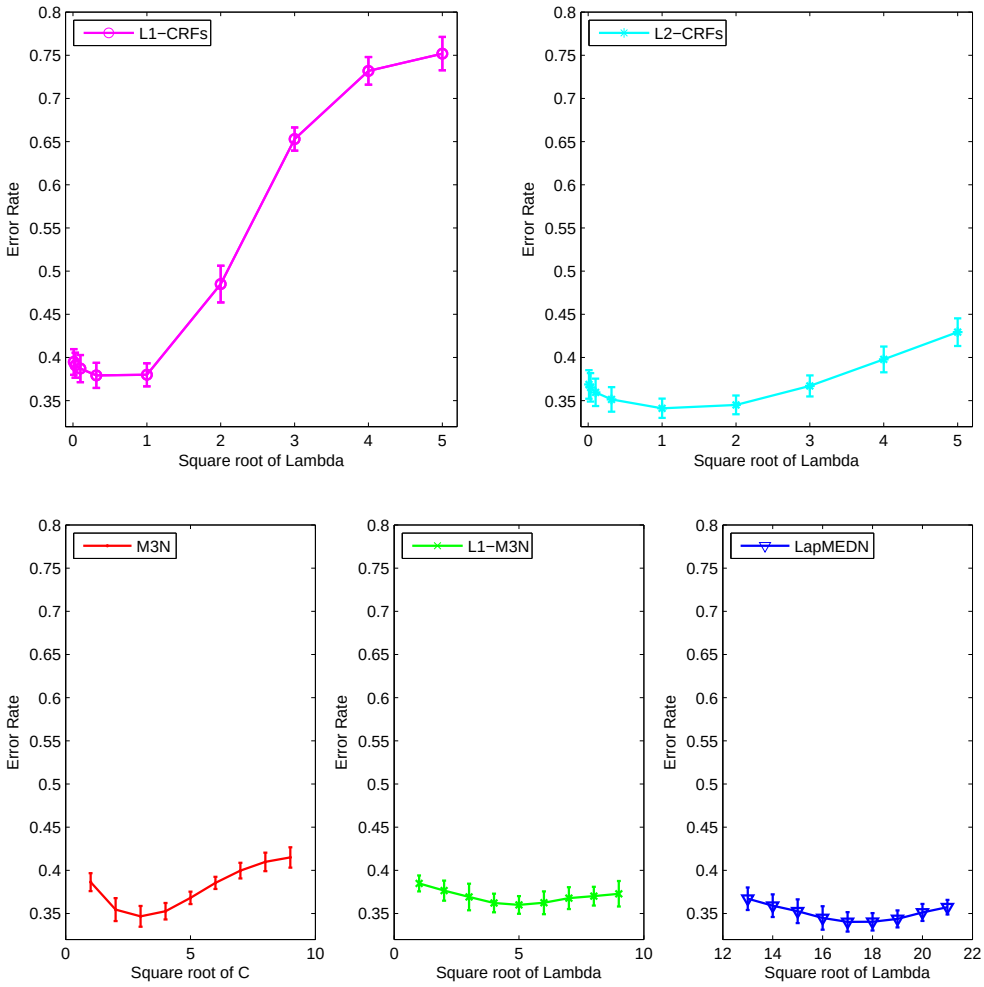


Figure 9: Error rates of different models on OCR100 with different regularization constants. The regularization constant is the parameter C for M^{3}N, and for all the other models, it is the parameter λ. From left to right, the regularization constants for the two regularized CRFs (above plot) are 0.0001, 0.001, 0.01, 0.1, 1, 4, 9, 16, and 25; for M^3N and LapMEDN, the regularization constants are $k^2,\ 1 \leq k \leq 9$; and for L_1-M^{3}N, the constants are $k^2,\ 13 \leq k \leq 21$.

potential advantages of L_1-norm regularized M^{3}N, which are consistently better than the L_1-CRFs. Furthermore, as we shall see later, L_1-M^{3}N is more stable than the L_1-CRFs.

7.3 Sensitivity to Regularization Constants

Figure 9 shows the error rates of the models in question on the data set OCR100 over different magnitudes of the regularization constants. For M^{3}N, the regularization constant is the parameter

C, and for all the other models, the regularization constant is the parameter λ. When the λ changes, the parameter C in LapMEDN and L_1-M^{3}N is fixed at the unit 1.

From the results, we can see that the L_1-CRFs are quite sensitive to the regularization constants. However, L_2-CRFs, M^{3}N, L_1-M^{3}N and LapMEDN are much less sensitive. LapMEDN and L_1-M^{3}N are the most stable models. The stability of LapMEDN is due to the posterior weighting instead of hard-thresholding to set small weights to zero as in the L_1-CRFs. One interesting observation is that the max-margin based L_1-M^{3}N is much more stable compared to the L_1-norm regularized CRFs. One possible reason is that like LapMEDN, L_1-M^{3}N enjoys both the primal and dual sparsity, which makes it less sensitive to outliers; whereas the L_1-CRF is only primal sparse.

7.4 Real-World Web Data Extraction

The last experiments are conducted on the real world web data extraction as extensively studied by Zhu et al. (2008a). Web data extraction is a task to identify interested information from web pages. Each sample is a data record or an entire web page which is represented as a set of HTML elements. One striking characteristic of web data extraction is that various types of structural dependencies between HTML elements exist, e.g. the HTML tag tree or the Document Object Model (DOM) structure is itself hierarchical. In the work of Zhu et al. (2008a), hierarchical CRFs are shown to have great promise and achieve better performance than flat models like linear-chain CRFs (Lafferty et al., 2001). One method to construct a hierarchical model is to first use a parser to construct a so called vision tree. Then, based on the vision tree, a hierarchical model can be constructed accordingly to extract the interested attributes, e.g. a product's name, image, price, description, etc. See the paper (Zhu et al., 2008a) for an example of the vision tree and the corresponding hierarchical model. In such a hierarchical extraction model, inner nodes are useful to incorporate long distance dependencies, and the variables at one level are refinements of the variables at upper levels.

In these experiments,[6] we identify product items for sale on the Web. For each product item, four attributes—*Name*, *Image*, *Price*, and *Description* are extracted. We use the data set that is built with web pages generated by 37 different templates (Zhu et al., 2008a). For each template, there are 5 pages for training and 10 for testing. We evaluate all the methods on the *record level*, that is, we assume that data records are given, and we compare different models on the accuracy of extracting attributes in the given records. In the 185 training pages, there are 1585 data records in total; in the 370 testing pages, 3391 data records are collected. As for the evaluation criteria, we use the two comprehensive measures, that is, average F1 and block instance accuracy. As defined by Zhu et al. (2008a), average F1 is the average value of the F1 scores of the four attributes, and block instance accuracy is the percent of data records whose *Name*, *Image*, and *Price* are all correctly identified.

We randomly select $m = 5, 10, 15, 20, 30, 40, \text{or}, 50$ percent of the training records as training data, and test on all the testing records. For each m, 10 independent experiments were conducted and the average performance is summarized in Figure 10. From the results, we can see that all: first, the models (especially the max-margin models, that is, M^{3}N, L_1-M^{3}N, and LapMEDN) with regularization (i.e., L_1-norm, L_2-norm, or the entropic regularization of LapMEDN) can significantly outperform the un-regularized CRFs. Second, the max-margin models generally outperform the conditional likelihood-based models (i.e., CRFs, L_2-CRFs, and L_1-CRFs). Third, the LapMEDN perform comparably with the L_1-M^{3}N, which enjoys both dual and primal sparsity as the LapMEDN, and

6. These experiments are slightly different from those by Zhu et al. (2008a). Here, we introduce more general feature functions based on the content and visual features as used by Zhu et al. (2008a).

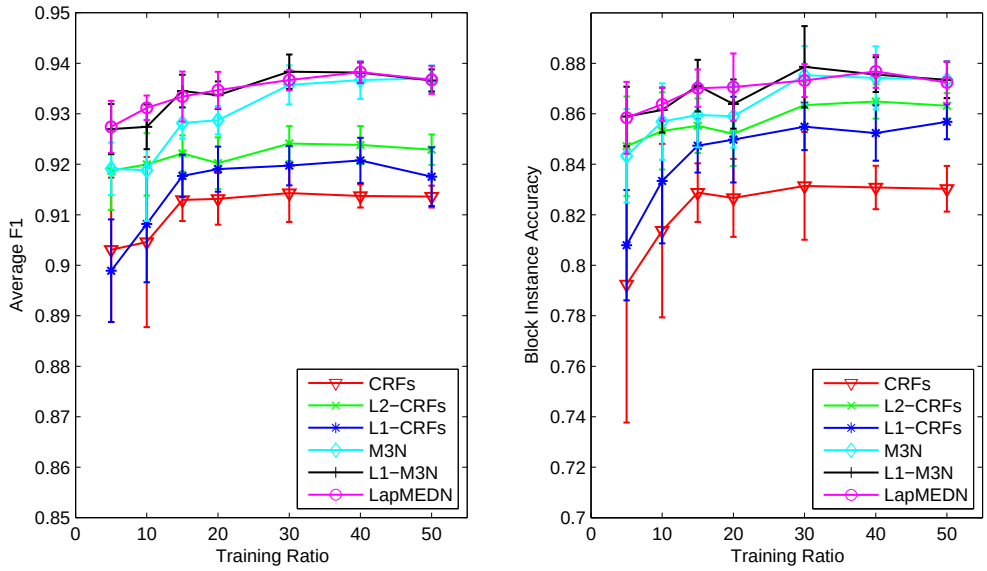


Figure 10: The average F1 values and block instance accuracy on web data extraction with different number of training data.

outperforms all other models, especially when the number of training data is small. Finally, as in the previous experiments on OCR data, the L_1-M^{3}N generally outperforms the L_1-CRFs, which suggests the potential promise of the max-margin based L_1-M^{3}N.

8. Related Work

Our work is motivated by the maximum entropy discrimination (MED) method proposed by Jaakkola et al. (1999), which integrates SVM and entropic regularization to obtain an averaging maximum margin model for classification. The MaxEnDNet model presented is essentially a structured version of MED built on M^{3}N—the so called "structured SVM". As we presented in this paper, this extension leads to a substantially more flexible and powerful new paradigm for structured discriminative learning and prediction, which enjoys a number of advantages such as model averaging, primal and dual sparsity, accommodation of latent generative structures, but at the same time raises new algorithmic challenges in inference and learning.

Related to our approach, a sparse Bayesian learning framework has been proposed to find sparse and robust solutions to regression and classification. One example along this line is the relevance vector machine (RVM) (Tipping, 2001). The RVM was proposed based on SVM. But unlike SVM which directly optimizes on the margins, RVM defines a likelihood function from the margins with a Gaussian distribution for regression and a logistic sigmoid link function for classification and then does *type-II maximum likelihood* estimation, that is, RVM maximizes the *marginal likelihood*. Although called *sparse Bayesian learning* (Figueiredo, 2001; Eyheramendy et al., 2003), as shown by Kaban (2007) the sparsity is actually due to the MAP estimation. The similar ambiguity of RVM is justified by Wipf et al. (2003). Unlike these approaches, we adhere to a full Bayesian-style principle and learn a distribution of predictive models by optimizing a generalized maximum entropy under a set of the *expected* margin constraints. By defining likelihood functions with margins, similar

Bayesian interpretations of both binary and multi-class SVM were studied by Sollich (2002) and Zhang and Jordan (2006).

The hierarchical interpretation of the Laplace prior has been explored in a number of contexts in the literature. Based on this interpretation, a Jeffrey's non-informative second-level hyper-prior was proposed by Figueiredo (2001), with an EM algorithm developed to find the MAP estimate. The advantage of the Jeffrey's prior is that it is parameter-free. But as shown by Eyheramendy et al. (2003) and Kaban (2007), usually no advantage is achieved by using the Jeffrey's hyper-prior over the Laplace prior. A gamma hyper-prior was used by Tipping (2001) in place of the second-level exponential as in the hierarchical interpretation of the Laplace prior.

To encourage sparsity in SVM, two strategies have been used. The first one is to replace the L_2-norm by an L_1-norm of the weights (Bennett and Mangasarian, 1992; Zhu et al., 2004). The second strategy is to explicitly add a cardinality constraint on the weights. This will lead to a hard non-convex optimization problem; thus relaxations must be applied (Chan et al., 2007). Under the maximum entropy discrimination models, feature selection was studied by Jebara and Jaakkola (2000) by introducing a set of structural variables. Recently, a smooth posterior shrinkage effect was shown by Jebara (2009), which is similar to our entropic regularization effect. However, an analysis of their connections and differences is still not obvious.

Although the parameter distribution $p(\mathbf{w})$ in Theorem 2 has a similar form as that of the Bayesian Conditional Random Fields (BCRFs) (Qi et al., 2005), MaxEnDNet is fundamentally different from BCRFs as we have stated. Dredze et al. (2008) present an interesting confidence-weighted linear classification method, which automatically estimates the mean and variance of model parameters in online learning. The procedure is similar to (but indeed different from) our variational Bayesian method of Laplace MaxEnDNet.

Finally, some of the results shown in this paper can be also found in our recent conference papers (Zhu et al., 2008b; Zhu and Xing, 2009).

9. Conclusions and Future Work

To summarize, we have presented a general theory of maximum entropy discrimination Markov networks for structured input/output learning and prediction. This formalism offers a formal paradigm for integrating both generative and discriminative principles and the Bayesian regularization techniques for learning structured prediction models. It subsumes popular methods such as support vector machines, maximum entropy discrimination models (Jaakkola et al., 1999), and maximum margin Markov networks as special cases, and therefore inherits all the merits of these techniques.

The MaxEnDNet model offers a number of important advantages over conventional structured prediction methods, including: 1) model averaging, which leads to a PAC-Bayesian bound on generalization error; 2) entropic regularization over max-margin learning, which can be leveraged to learn structured prediction models that are simultaneously primal and dual sparse; and 3) latent structures underlying the structured input/output variables, which enables better incorporation of domain knowledge in model design and semi-supervised learning based on partially labeled data. In this paper, we have discussed in detail the first two aspects, and the third aspect was explored in (Zhu et al., 2008c). We have also shown that certain instantiations of the MaxEnDNet model, such as the LapMEDN that achieves primal and dual sparsity, can be efficiently trained based on an iterative optimization scheme that employs existing techniques such as the variational Bayes approximation and the convex optimization procedures that solve the standard M^3N. We demonstrated that on syn-

thetic data the LapMEDN can recover the sparse model as well as the sparse L_1-regularized MAP estimation, and on real data sets LapMEDN can achieve superior performance.

Overall, we believe that the MaxEnDNet model can be extremely general and adaptive, and it offers a promising new framework for building more flexible, generalizable, and large scale structured prediction models that enjoy the benefits from both generative and discriminative modeling principles. While exploring novel instantiations of this model will be an interesting direction to pursue, development of more efficient learning algorithms, formulation of tighter but easy to solve convex relaxations, and adapting this model to challenging applications such as statistical machine translation, and structured associations of genome markers to complex disease traits could also lead to fruitful results. Finally, the basic principle of MaxEnDNet can be generalized to directed graphical models. The MedLDA model (Zhu et al., 2009a) for discriminative topic modeling represents our first successful attempt along this direction.

Acknowledgments

Jun Zhu was with the Department of Computer Science and Technology, Tsinghua University. This work was done while Jun Zhu was visiting the SAILING Lab directed by Eric Xing at Carnegie Mellon University, under a visiting scholarship sponsored by the National Natural Science Foundation of China. The authors would like to thank Andrew Bagnell for sharing their implementation of the sub-gradient algorithm, Ivor Tsang and André Martins for inspiring discussions. Eric Xing is supported by NSF grants CCF-0523757, DBI-0546594, IIS-0713379, DBI- 0640543, and a Sloan Research Fellowship in Computer Science. Jun Zhu was supported by the National Natural Science Foundation of China, Grant. No. 60321002; the National Key Foundation R&D Projects, Grant No. 2004CB318108 and 2007CB311003; and Microsoft Fellowship.

Appendix A. L_1-M^{3}N and its Lagrange-Dual

Based on the L_1-norm regularized SVM (Zhu et al., 2004; Bennett and Mangasarian, 1992), a straightforward formulation of L_1-M^{3}N is as follows,

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

where $\|.\|$ is the L_1-norm. $\Delta\mathbf{f}_i(\mathbf{y}) = \mathbf{f}(\mathbf{x}^i,\mathbf{y}^i) - \mathbf{f}(\mathbf{x}^i,\mathbf{y})$, and $\Delta\ell_i(\mathbf{y})$ is a loss function. Another equivalent formulation[7] is as follows:

$$\begin{aligned}&\min_{\mathbf{w},\xi} \ C\sum_{i=1}^{N}\xi_i \\ &\text{s.t.} \ \ \begin{cases} \|\mathbf{w}\| \leq \lambda \\ \mathbf{w}^\top \Delta\mathbf{f}_i(\mathbf{y}) \geq \Delta\ell_i(\mathbf{y}) - \xi_i;\ \xi_i \geq 0,\ \forall i,\ \forall \mathbf{y} \neq \mathbf{y}^i. \end{cases}\end{aligned}$$

7. See Taskar et al. (2006) for the transformation technique.

To derive the convex dual problem, we introduce a dual variable $\alpha_i(\mathbf{y})$ for each constraint in the former formulation and form the Lagrangian as follows,

$$L(\alpha, \mathbf{w}, \xi) = \frac{1}{2}\|\mathbf{w}\| + C\sum_{i=1}^{N}\xi_i - \sum_{i,\mathbf{y}\neq\mathbf{y}^i}\alpha_i(\mathbf{y})(\mathbf{w}^\top\Delta\mathbf{f}_i(\mathbf{y}) - \Delta\ell_i(\mathbf{y}) + \xi_i).$$

By definition, the Lagrangian dual is,

$$\begin{aligned} L^\star(\alpha) &= \inf_{\mathbf{w},\xi} L(\alpha, \mathbf{w}, \xi) \\ &= \inf_{\mathbf{w}}\Big[\frac{1}{2}\|\mathbf{w}\| - \sum_{i,\mathbf{y}\neq\mathbf{y}^i}\alpha_i(\mathbf{y})\mathbf{w}^\top\Delta\mathbf{f}_i(\mathbf{y})\Big] + \inf_{\xi}\Big[C\sum_{i=1}^{N}\xi_i - \sum_{i,\mathbf{y}\neq\mathbf{y}^i}\alpha_i(\mathbf{y})\xi_i\Big] + \ell \\ &= -\sup_{\mathbf{w}}\Big[\mathbf{w}^\top(\sum_{i,\mathbf{y}\neq\mathbf{y}^i}\alpha_i(\mathbf{y})\Delta\mathbf{f}_i(\mathbf{y})) - \frac{1}{2}\|\mathbf{w}\|\Big] - \sup_{\xi}\Big[\sum_{i,\mathbf{y}\neq\mathbf{y}^i}\alpha_i(\mathbf{y})\xi_i - C\sum_{i=1}^{N}\xi_i\Big] + \ell, \end{aligned}$$

where $\ell = \sum_{i,\mathbf{y}\neq\mathbf{y}^i}\alpha_i(\mathbf{y})\Delta\ell_i(\mathbf{y})$.

Again, by definition, the first term on the right-hand side is the convex conjugate of $\phi(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|$ and the second term is the conjugate of $U(\xi) = C\sum_{i=1}^{N}\xi_i$. It is easy to show that,

$$\phi^\star(\alpha) = \mathbb{I}_\infty(|\sum_{i,\mathbf{y}\neq\mathbf{y}^i}\alpha_i(\mathbf{y})\Delta\mathbf{f}_i^k(\mathbf{y})| \leq \frac{1}{2},\ \forall 1 \leq k \leq K),$$

and

$$U^\star(\alpha) = \mathbb{I}_\infty(\sum_{\mathbf{y}\neq\mathbf{y}^i}\alpha_i(\mathbf{y}) \leq C,\ \forall i),$$

where as defined before $\mathbb{I}_\infty(\cdot)$ is an indicator function that equals zero when its argument is true and infinity otherwise. $\Delta\mathbf{f}_i^k(\mathbf{y}) = f_k(\mathbf{x}^i, \mathbf{y}^i) - f_k(\mathbf{x}^i, \mathbf{y})$.

Therefore, we get the dual problem as follows,

$$\begin{aligned} \max_{\alpha}\ & \sum_{i,\mathbf{y}\neq\mathbf{y}^i}\alpha_i(\mathbf{y})\Delta\ell_i(\mathbf{y}) \\ \text{s.t. }\ & |\sum_{i,\mathbf{y}\neq\mathbf{y}^i}\alpha_i(\mathbf{y})\Delta\mathbf{f}_i^k(\mathbf{y})| \leq \frac{1}{2},\ \forall k \\ & \sum_{\mathbf{y}\neq\mathbf{y}^i}\alpha_i(\mathbf{y}) \leq C,\ \forall i. \end{aligned}$$

Appendix B. Proofs of Theorems and Corollaries

In this section, we prove the theorems and corollaries.

B.1 Proof of Theorem 2

Proof As we have stated, P1 is a convex program and satisfies the Slater's condition. To compute its convex dual, we introduce a non-negative dual variable $\alpha_i(\mathbf{y})$ for each constraint in $\mathcal{F}_1$ and another

non-negative dual variable c for the normalization constraint $\int p(\mathbf{w})\,\mathrm{d}\mathbf{w} = 1$. This gives rise to the following Lagrangian:

$$\begin{aligned} L(p(\mathbf{w}),\xi,\alpha,c) \quad &= KL(p(\mathbf{w})||p_0(\mathbf{w})) + U(\xi) + c(\int p(\mathbf{w})\,\mathrm{d}\mathbf{w} - 1) \\ &\quad - \sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y})(\int p(\mathbf{w})[\Delta F_i(\mathbf{y};\mathbf{w}) - \Delta\ell_i(\mathbf{y})]\,\mathrm{d}\mathbf{w} + \xi_i). \end{aligned}$$

The Lagrangian dual function is defined as $L^\star(\alpha,c) \triangleq \inf_{p(\mathbf{w});\xi} L(p(\mathbf{w}),\xi,\alpha,c)$. Taking the derivative of L w.r.t $p(\mathbf{w})$, we get,

$$\frac{\partial L}{\partial p(\mathbf{w})} = 1 + c + \log\frac{p(\mathbf{w})}{p_0(\mathbf{w})} - \sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y})[\Delta F_i(\mathbf{y};\mathbf{w}) - \Delta\ell_i(\mathbf{y})].$$

Setting the derivative to zero, we get the following expression of distribution $p(\mathbf{w})$,

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

where $Z(\alpha) \triangleq \int p_0(\mathbf{w}) \exp\{\sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y})[\Delta F_i(\mathbf{y};\mathbf{w}) - \Delta\ell_i(\mathbf{y})]\}\,\mathrm{d}\mathbf{w}$ is a normalization constant and $c = -1 + \log Z(\alpha)$.

Substituting $p(\mathbf{w})$ into $L^\star$, we obtain,

$$\begin{aligned} L^\star(\alpha,c) \quad &= \quad \inf_{p(\mathbf{w});\xi} (-\log Z(\alpha) + U(\xi) - \sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y})\xi_i) \\ &= \quad -\log Z(\alpha) + \inf_{\xi} (U(\xi) - \sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y})\xi_i) \\ &= \quad -\log Z(\alpha) - \sup_{\xi} (\sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y})\xi_i - U(\xi)) \\ &= \quad -\log Z(\alpha) - U^\star(\alpha), \end{aligned}$$

which is the objective in the dual problem D1. The $\{\alpha_i(\mathbf{y})\}$ derived from D1 lead to the optimum $p(\mathbf{w})$ according to Equation (3). ■

B.2 Proof of Theorem 3

Proof Replacing $p_0(\mathbf{w})$ and $\Delta F_i(\mathbf{y};\mathbf{w})$ in Equation (3) with $\mathcal{N}(\mathbf{w}|0,I)$ and $\mathbf{w}^\top \Delta\mathbf{f}_i(\mathbf{y})$ respectively, we can obtain the following closed-form expression of the $Z(\alpha)$ in $p(\mathbf{w})$:

$$\begin{aligned} Z(\alpha) \quad &\triangleq \int \mathcal{N}(\mathbf{w}|0,I) \exp\Big\{\sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y})[\mathbf{w}^\top \Delta\mathbf{f}_i(\mathbf{y}) - \Delta\ell_i(\mathbf{y})]\Big\}\,\mathrm{d}\mathbf{w} \\ &= \int (2\pi)^{-\frac{K}{2}} \exp\Big\{-\frac{1}{2}\mathbf{w}^\top\mathbf{w} + \sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y})[\mathbf{w}^\top \Delta\mathbf{f}_i(\mathbf{y}) - \Delta\ell_i(\mathbf{y})]\Big\}\,\mathrm{d}\mathbf{w} \\ &= \exp\Big(-\sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y})\Delta\ell_i(\mathbf{y}) + \frac{1}{2}\|\sum_{i,\mathbf{y}\neq\mathbf{y}^i} \alpha_i(\mathbf{y})\Delta\mathbf{f}_i(\mathbf{y})\|^2\Big). \end{aligned}$$

Substituting the normalization factor into the general dual problem D1, we get the dual problem of Gaussian MaxEnDNet. As we have stated, the constraints $\sum_{\mathbf{y}\neq\mathbf{y}^i}\alpha_i(\mathbf{y}) = C$ are due to the conjugate of $U(\xi) = C\sum_i \xi_i$.

For prediction, again replacing $p_0(\mathbf{w})$ and $\Delta F_i(\mathbf{y};\mathbf{w})$ in Equation (3) with $\mathcal{N}(\mathbf{w}|0,I)$ and $\mathbf{w}^\top\Delta\mathbf{f}_i(\mathbf{y})$ respectively, we can get $p(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mu,I)$, where $\mu = \sum_{i,\mathbf{y}\neq\mathbf{y}^i}\alpha_i(\mathbf{y})\Delta\mathbf{f}_i(\mathbf{y})$. Substituting $p(\mathbf{w})$ into the predictive function h_1, we can get $h_1(\mathbf{x}) = \arg\max_{\mathbf{y}\in\mathcal{Y}(\mathbf{x})}\mu^\top\mathbf{f}(\mathbf{x},\mathbf{y}) = (\sum_{i,\mathbf{y}\neq\mathbf{y}^i}\alpha_i(\mathbf{y})\Delta\mathbf{f}_i(\mathbf{y}))^\top\mathbf{f}(\mathbf{x},\mathbf{y})$, which is identical to the prediction rule of the standard M^{3}N (Taskar et al., 2003) because the dual parameters are achieved by solving the same dual problem. ■

B.3 Proof of Corollary 4

Proof Suppose $(p^\star(\mathbf{w}),\xi^\star)$ is the optimal solution of P1, then we have: for any $(p(\mathbf{w}),\xi),\ p(\mathbf{w}) \in \mathcal{F}_1$ and $\xi \geq 0$,

$$KL(p^\star(\mathbf{w})||p_0(\mathbf{w})) + U(\xi^\star) \leq KL(p(\mathbf{w})||p_0(\mathbf{w})) + U(\xi).$$

From Theorem 3, we conclude that the optimum predictive parameter distribution is $p^\star(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mu^\star,I)$. Since $p_0(\mathbf{w})$ is also normal, for any distribution $p(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mu,I)$[8] with several steps of algebra it is easy to show that $KL(p(\mathbf{w})|p_0(\mathbf{w})) = \frac{1}{2}\mu^\top\mu$. Thus, we can get: for any $(\mu,\xi),\ \mu \in \{\mu:\ \mu^\top\Delta\mathbf{f}_i(\mathbf{y}) \geq \Delta\ell_i(\mathbf{y}) - \xi_i,\ \forall i,\ \forall\mathbf{y}\neq\mathbf{y}^i\}$ and $\xi \geq 0$,

$$\frac{1}{2}(\mu^\star)^\top(\mu^\star) + U(\xi^\star) \leq \frac{1}{2}\mu^\top\mu + U(\xi^\star),$$

which means the mean of the optimum posterior distribution under a Gaussian MaxEnDNet is achieved by solving a primal problem as stated in the Corollary. ■

B.4 Proof of Corollary 6

Proof The proof follows the same structure as the above proof of Corollary 4. Here, we only present the derivation of the KL-divergence under the Laplace MaxEnDNet.

Theorem 2 shows that the general posterior distribution is $p(\mathbf{w}) = \frac{1}{Z(\alpha)}p_0(\mathbf{w})\exp(\mathbf{w}^\top\eta - \sum_{i,\mathbf{y}\neq\mathbf{y}^i}\alpha_i(\mathbf{y})\Delta\ell_i(\mathbf{y}))$ and $Z(\alpha) = \exp(-\sum_{i,\mathbf{y}\neq\mathbf{y}^i}\alpha_i(\mathbf{y})\Delta\ell_i(\mathbf{y}))\prod_{k=1}^K\frac{\lambda}{\lambda-\eta_k^2}$ for the Laplace MaxEnDNet as shown in Equation (5). Use the definition of KL-divergence and we can get:

$$KL(p(\mathbf{w})|p_0(\mathbf{w})) = \langle\mathbf{w}\rangle_p^\top\eta - \sum_{k=1}^K\log\frac{\lambda}{\lambda-\eta_k^2} = \sum_{k=1}^K\mu_k\eta_k - \sum_{k=1}^K\log\frac{\lambda}{\lambda-\eta_k^2},$$

Corollary 7 shows that $\mu_k = \frac{2\eta_k}{\lambda-\eta_k^2},\ \forall 1\leq k\leq K$. Thus, we get $\frac{\lambda}{\lambda-\eta_k^2} = \frac{\lambda\mu_k}{2\eta_k}$ and a set of equations: $\mu_k\eta_k^2 + 2\eta_k - \lambda\mu_k = 0,\ \forall 1\leq k\leq K$. To solve these equations, we consider two cases. First, if $\mu_k = 0$, then $\eta_k = 0$. Second, if $\mu_k \neq 0$, then we can solve the quadratic equation to get

8. Although $\mathcal{F}_1$ is much richer than the set of normal distributions with an identity covariance matrix, Theorem 3 shows that the solution is a restricted normal distribution. Thus, it suffices to consider only these normal distributions in order to learn the mean of the optimum distribution. The similar argument applies to the proof of Corollary 6.

η_k: $\eta_k = \frac{-1\pm\sqrt{1+\lambda\mu_k^2}}{\mu_k}$. The second solution includes the first one since we can show that when $\mu_k \to 0$, $\frac{-1\pm\sqrt{1+\lambda\mu_k^2}}{\mu_k} \to 0$ by using the *L'Hospital's Rule*. Thus, we get:

$$\mu_k\eta_k = -1 \pm \sqrt{\lambda\mu_k^2+1}.$$

Since $\eta_k^2 < \lambda$ (otherwise the problem is not bounded), $\mu_k\eta_k$ is always positive. Thus, only the solution $\mu_k\eta_k = -1+\sqrt{1+\lambda\mu_k^2}$ is feasible. So, we get:

$$\frac{\lambda}{\lambda-\eta_k^2} = \frac{\lambda\mu_k^2}{2(\sqrt{\lambda\mu_k^2+1}-1)} = \frac{\sqrt{\lambda\mu_k^2+1}+1}{2},$$

and

$$\begin{aligned} KL(p(\mathbf{w})|p_0(\mathbf{w})) &= \sum_{k=1}^{K}\Big(\sqrt{\lambda\mu_k^2+1} - \log\frac{\sqrt{\lambda\mu_k^2+1}+1}{2}\Big) - K \\ &= \sqrt{\lambda}\sum_{k=1}^{K}\Big(\sqrt{\mu_k^2+\frac{1}{\lambda}} - \frac{1}{\sqrt{\lambda}}\log\frac{\sqrt{\lambda\mu_k^2+1}+1}{2}\Big) - K. \end{aligned}$$

Applying the same arguments as in the above proof of Corollary 4 and using the above result of the KL-divergence, we get the problem in Corollary 6, where the constant $-K$ is ignored. The margin constraints defined with the mean μ are due to the linearity assumption of the discriminant functions. ■

B.5 Proof of Theorem 8

We follow the same structure as the proof of PAC-Bayes bound for binary classifier (Langford et al., 2001) and employ the similar technique to generalize to multi-class problems (Schapire et al., 1998). Recall that the output space is $\mathcal{Y}$, and the base discriminant function is $F(\,\cdot\,;\mathbf{w}) \in \mathcal{H} : \mathcal{X}\times\mathcal{Y} \to [-c,c]$, where $c>0$ is a constant. Our averaging model is specified by $h(\mathbf{x},\mathbf{y}) = \langle F(\mathbf{x},\mathbf{y};\mathbf{w})\rangle_{p(\mathbf{w})}$. We define the margin of an example $(\mathbf{x},\mathbf{y})$ for such a function h as,

$$M(h,\mathbf{x},\mathbf{y}) = h(\mathbf{x},\mathbf{y}) - \max_{\mathbf{y}'\neq\mathbf{y}} h(\mathbf{x},\mathbf{y}'). \tag{12}$$

Thus, 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 independently and randomly drawn from Q. With these definitions, we have the PAC-Bayes theorem. For easy reading, we copy the theorem in the following:

Theorem 8 (PAC-Bayes Bound of MaxEnDNet) *Let p_0 be any continuous probability distribution over $\mathcal{H}$ and let $\delta\in(0,1)$. If $F(\,\cdot\,;\mathbf{w}) \in \mathcal{H}$ is bounded by $\pm c$ as above, then with probability*

at least $1-\delta$*, for a random sample* $\mathcal{D}$ *of* N *instances from* Q*, for every distribution* p *over* $\mathcal{H}$*, and for all margin thresholds* $\gamma > 0$*:*

$$\Pr_Q(M(h,\mathbf{x},\mathbf{y}) \leq 0) \leq \Pr_{\mathcal{D}}(M(h,\mathbf{x},\mathbf{y}) \leq \gamma) + O\Big(\sqrt{\frac{\gamma^{-2}KL(p||p_0)\ln(N|\mathcal{Y}|) + \ln N + \ln \delta^{-1}}{N}}\Big),$$

where $\Pr_Q(.)$ *and* $\Pr_{\mathcal{D}}(.)$ *represent the probabilities of events over the true distribution* Q*, and over the empirical distribution of* $\mathcal{D}$*, respectively.*

Proof Let m be any natural number. For every distribution p, we independently draw m base models (i.e., discriminant functions) $F_i \sim p$ at random. We also independently draw m variables $\mu_i \sim U([-c,c])$, where U denote the uniform distribution. We define the binary functions $g_i : \mathcal{X} \times \mathcal{Y} \rightarrow \{-c,+c\}$ by:

$$g_i(\mathbf{x},\mathbf{y};F_i,\mu_i) = 2cI(\mu_i < F_i(\mathbf{x},\mathbf{y})) - c.$$

With the F_i, μ_i, and g_i, we define $\mathcal{H}_m$ as,

$$\mathcal{H}_m = \{f : (\mathbf{x},\mathbf{y}) \mapsto \frac{1}{m}\sum_{i=1}^{m} g_i(\mathbf{x},\mathbf{y};F_i,\mu_i) | F_i \in \mathcal{H}, \mu_i \in [-c,c]\}.$$

We denote the distribution of f over the set $\mathcal{H}_m$ by p^m. For a fixed pair $(\mathbf{x},\mathbf{y})$, the quantities $g_i(\mathbf{x},\mathbf{y};F_i,\mu_i)$ are i.i.d bounded random variables with the mean:

$$\begin{aligned}
\langle g_i(\mathbf{x},\mathbf{y};F_i,\mu_i)\rangle_{F_i\sim p,\mu_i\sim U[-c,c]} &= \langle (+c)p[\mu_i \leq F_i(\mathbf{x},\mathbf{y})|F_i] + (-c)p[\mu_i > F_i(\mathbf{x},\mathbf{y})|F_i]\rangle_{F_i\sim p} \\
&= \langle \frac{1}{2c}c(c+F_i(\mathbf{x},\mathbf{y})) - \frac{1}{2c}c(c-F_i(\mathbf{x},\mathbf{y}))\rangle_{F_i\sim p} \\
&= h(\mathbf{x},\mathbf{y}).
\end{aligned}$$

Therefore, $\langle f(\mathbf{x},\mathbf{y})\rangle_{f\sim p^m} = h(\mathbf{x},\mathbf{y})$. Since $f(\mathbf{x},\mathbf{y})$ is the average over m i.i.d bounded variables, Hoeffding's inequality applies. Thus, for every $(\mathbf{x},\mathbf{y})$,

$$\Pr_{f\sim p^m}[f(\mathbf{x},\mathbf{y}) - h(\mathbf{x},\mathbf{y}) > \xi] \leq e^{-\frac{m}{2c^2}\xi^2}.$$

For any two events A and B, we have the inequality,

$$\Pr(A) = \Pr(A,B) + \Pr(A,\bar{B}) \leq \Pr(B) + \Pr(\bar{B}|A).$$

Thus, for any $\gamma > 0$ we have

$$\Pr_Q[M(h,\mathbf{x},\mathbf{y}) \leq 0] \leq \Pr_Q[M(f,\mathbf{x},\mathbf{y}) \leq \frac{\gamma}{2}] + \Pr_Q[M(f,\mathbf{x},\mathbf{y}) > \frac{\gamma}{2} | M(h,\mathbf{x},\mathbf{y}) \leq 0], \tag{13}$$

where the left hand side does not depend on f. We take the expectation w.r.t $f \sim p^m$ on both sides and can get

$$\Pr_Q[M(h,\mathbf{x},\mathbf{y}) \leq 0] \leq \quad \langle \Pr_Q[M(f,\mathbf{x},\mathbf{y}) \leq \frac{\gamma}{2}] \rangle_{f\sim p^m} + \langle \Pr_Q[M(f,\mathbf{x},\mathbf{y}) > \frac{\gamma}{2} | M(h,\mathbf{x},\mathbf{y}) \leq 0] \rangle_{f\sim p^m}. \tag{14}$$

Fix $h, \mathbf{x}$, and $\mathbf{y}$, and let $\mathbf{y}'$ achieve the margin in (12). Then, we get

$$M(h,\mathbf{x},\mathbf{y}) = h(\mathbf{x},\mathbf{y}) - h(\mathbf{x},\mathbf{y}'), \text{ and } M(f,\mathbf{x},\mathbf{y}) \leq f(\mathbf{x},\mathbf{y}) - f(\mathbf{x},\mathbf{y}').$$

With these two results, since $\langle f(\mathbf{x},\mathbf{y}) - f(\mathbf{x},\mathbf{y}') \rangle_{f\sim p^m} = h(\mathbf{x},\mathbf{y}) - h(\mathbf{x},\mathbf{y}')$, we can get

$$\begin{aligned}
\langle \Pr_Q[M(f,\mathbf{x},\mathbf{y}) > \frac{\gamma}{2} | M(h,\mathbf{x},\mathbf{y}) \leq 0] \rangle_{f\sim p^m} \quad &= \langle \Pr_{f\sim p^m}[M(f,\mathbf{x},\mathbf{y}) > \frac{\gamma}{2} | M(h,\mathbf{x},\mathbf{y}) \leq 0] \rangle_Q \\
&\leq \langle \Pr_{f\sim p^m}[f(\mathbf{x},\mathbf{y}) - f(\mathbf{x},\mathbf{y}') > \frac{\gamma}{2} | M(h,\mathbf{x},\mathbf{y}) \leq 0] \rangle_Q \\
&\leq \langle \Pr_{f\sim p^m}[f(\mathbf{x},\mathbf{y}) - f(\mathbf{x},\mathbf{y}') - M(h,\mathbf{x},\mathbf{y}) > \frac{\gamma}{2}] \rangle_Q \\
&\leq e^{-\frac{m\gamma^2}{32c^2}},
\end{aligned} \tag{15}$$

where the first two inequalities are due to the fact that if two events $A \subseteq B$, then $p(A) \leq p(B)$, and the last inequality is due to the Hoeffding's inequality.

Substitute (15) into (14), and we get,

$$\Pr_Q[M(h,\mathbf{x},\mathbf{y}) \leq 0] \leq \langle \Pr_Q[M(f,\mathbf{x},\mathbf{y}) \leq \frac{\gamma}{2}] \rangle_{f\sim p^m} + e^{-\frac{m\gamma^2}{32c^2}}. \tag{16}$$

Let p_0^m be a prior distribution on $\mathcal{H}_m$. p_0^m is constructed from p_0 over $\mathcal{H}$ exactly as p^m is constructed from p. Then, $KL(p^m||p_0^m) = mKL(p||p_0)$. By the PAC-Bayes theorem (McAllester, 1999), with probability at least $1-\delta$ over sample $\mathcal{D}$, the following bound holds for any distribution p,

$$\langle \Pr_Q[M(f,\mathbf{x},\mathbf{y}) \leq \frac{\gamma}{2}] \rangle_{f\sim p^m} \quad \leq \langle \Pr_{\mathcal{D}}[M(f,\mathbf{x},\mathbf{y}) \leq \frac{\gamma}{2}] \rangle_{f\sim p^m} + \sqrt{\frac{mKL(p||p_0) + \ln N + \ln \delta^{-1} + 2}{2N-1}}. \tag{17}$$

By the similar statement as in (13), for every $f \in \mathcal{H}_m$ we have,

$$\Pr_{\mathcal{D}}[M(f,\mathbf{x},\mathbf{y}) \leq \frac{\gamma}{2}] \leq \Pr_{\mathcal{D}}[M(h,\mathbf{x},\mathbf{y}) \leq \gamma] + \Pr_{\mathcal{D}}[M(f,\mathbf{x},\mathbf{y}) \leq \frac{\gamma}{2} | M(h,\mathbf{x},\mathbf{y}) > \gamma].$$

Taking the expectation at both sides w.r.t $f \sim p^m$, we can get

$$\begin{aligned}\langle \Pr_{\mathcal{D}}[M(f,\mathbf{x},\mathbf{y}) \le \frac{\gamma}{2}]\rangle_{f\sim p^m} \quad &\le \Pr_{\mathcal{D}}[M(h,\mathbf{x},\mathbf{y}) \le \gamma] \\ &+\langle \Pr_{\mathcal{D}}[M(f,\mathbf{x},\mathbf{y}) \le \frac{\gamma}{2} | M(h,\mathbf{x},\mathbf{y}) > \gamma]\rangle_{f\sim p^m}, \end{aligned} \tag{18}$$

where the second term at the right hand side equals to $\langle \Pr_{f\sim p^m}[M(f,\mathbf{x},\mathbf{y}) \le \frac{\gamma}{2} | M(h,\mathbf{x},\mathbf{y}) > \gamma]\rangle_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}$ by exchanging the orders of expectations, and we can get

$$\begin{aligned}\Pr_{f\sim p^m}\left[M(f,\mathbf{x},\mathbf{y}) \le \frac{\gamma}{2} | M(h,\mathbf{x},\mathbf{y}) > \gamma\right] \quad &= \Pr_{f\sim p^m}\left[\exists \mathbf{y}' \ne \mathbf{y} : \Delta f(\mathbf{x},\mathbf{y}') \le \frac{\gamma}{2} | \forall \mathbf{y}' \ne \mathrm{y} : \Delta h(\mathbf{x},\mathbf{y}') > \gamma\right] \\ &\le \Pr_{f\sim p^m}\left[\exists \mathbf{y}' \ne \mathbf{y} : \Delta f(\mathbf{x},\mathbf{y}') \le \frac{\gamma}{2} | \Delta h(\mathbf{x},\mathbf{y}') > \gamma\right] \\ &\le \sum_{\mathbf{y}'\ne\mathbf{y}} \Pr_{f\sim p^m}\left[\Delta f(\mathbf{x},\mathbf{y}') \le \frac{\gamma}{2} | \Delta h(\mathbf{x},\mathbf{y}') > \gamma\right] \\ &\le (|\mathcal{Y}|-1)e^{-\frac{m\gamma^2}{32c^2}}, \end{aligned} \tag{19}$$

where we use $\Delta f(\mathbf{x},\mathbf{y}')$ to denote $f(\mathbf{x},\mathbf{y}) - f(\mathbf{x},\mathbf{y}')$, and use $\Delta h(\mathbf{x},\mathbf{y}')$ to denote $h(\mathbf{x},\mathbf{y}) - h(\mathbf{x},\mathbf{y}')$.

Put (16), (17), (18), and (19) together, then we get following bound holding for any fixed m and $\gamma > 0$,

$$\Pr_Q[M(h,\mathbf{x},\mathbf{y}) \le 0] \quad \le \Pr_{\mathcal{D}}[M(h,\mathbf{x},\mathbf{y}) \le \gamma] + |\mathcal{Y}|e^{-\frac{m\gamma^2}{32c^2}} + \sqrt{\frac{mKL(p||p_0) + \ln N + \ln\delta^{-1} + 2}{2N-1}}.$$

To finish the proof, we need to remove the dependence on m and γ. This can be done by applying the union bound. By the definition of f, it is obvious that if $f \in \mathcal{H}_m$ then $f(\mathbf{x},\mathbf{y}) \in \{(2k-m)c/m : k = 0,1,\ldots,m\}$. Thus, even though γ can be any positive value, there are no more than $m+1$ events of the form $\{M(f,\mathbf{x},\mathbf{y}) \le \gamma/2\}$. Since only the application of PAC-Bayes theorem in (17) depends on (m,γ) and all the other steps are true with probability one, we just need to consider the union of countably many events. Let $\delta_{m,k} = \delta/(m(m+1)^2)$, then the union of all the possible events has a probability at most $\sum_{m,k}\delta_{m,k} = \sum_m (m+1)\delta/(m(m+1)^2) = \delta$. Therefore, with probability at least $1-\delta$ over random samples of $\mathcal{D}$, the following bound holds for all m and all $\gamma > 0$,

$$\begin{aligned}\Pr_Q[M(h,\mathbf{x},\mathbf{y}) \le 0] - \Pr_{\mathcal{D}}[M(h,\mathbf{x},\mathbf{y}) \le \gamma] \quad &\le |\mathcal{Y}|e^{-\frac{m\gamma^2}{32c^2}} + \sqrt{\frac{mKL(p||p_0) + \ln N + \ln\delta_{m,k}^{-1} + 2}{2N-1}} \\ &\le |\mathcal{Y}|e^{-\frac{m\gamma^2}{32c^2}} + \sqrt{\frac{mKL(p||p_0) + \ln N + 3\ln\frac{m+1}{\delta} + 2}{2N-1}}. \end{aligned}$$

Setting $m = \lceil 16c^2\gamma^{-2}\ln\frac{N|\mathcal{Y}|^2}{KL(p||p_0)+1}\rceil$ gives the results in the theorem. ■

References

Yasemin Altun, Ioannis Tsochantaridis, and Thomas Hofmann. Hidden Markov support vector machines. In *International Conference on Machine Learning*, 2003.

Yasemin Altun, David McAllester, and Mikhail Belkin. Maximum margin semi-supervised learning for structured variables. In *Advances in Neural Information Processing Systems*, 2006.

Galen Andrew and Jianfeng Gao. Scalable training of l_1-regularized log-linear models. In *International Conference on Machine Learning*, 2007.

David F. Andrews and Colin L. Mallows. Scale mixtures of normal distributions. *Journal of the Royal Statistical Society*, B(1):99–102, 1974.

Peter L. Bartlett, Michael Collins, Ben Taskar, and David McAllester. Exponentiated gradient algorithms for larg-margin structured classification. In *Advances in Neural Information Processing Systems*, 2004.

Kristin P. Bennett and O. L. Mangasarian. Robust linear programming discrimination of two linearly inseparable sets. *Optim. Methods Softw*, (1):23–34, 1992.

Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

Ulf Brefeld and Tobias Scheffer. Semi-supervised learning for structured output variables. In *International Conference on Machine Learning*, 2006.

Antoni B. Chan, Nuno Vasconcelos, and Gert R. G. Lanckriet. Direct convex relaxations of sparse SVM. In *International Conference on Machine Learning*, 2007.

Mark Dredze, Koby Crammer, and Fernando Pereira. Confidence-weighted linear classification. In *International Conference on Machine Learning*, 2008.

John Duchi, Shai Shalev-Shwartz, Yoram Singer, and Tushar Chandra. Efficient projection onto the l_1-ball for learning in high dimensions. In *International Conference on Machine Learning*, 2008.

Mirosla Dudík, Steven J. Phillips, and Robert E. Schapire. Maximum entropy density estimation with generalized regularization and an application to species distribution modeling. *Journal of Machine Learning Research*, (8):1217–1260, 2007.

Susana Eyheramendy, Alexander Genkin, Wen-Hua Ju, David D. Lewis, and David Madiagan. Sparse Bayesian classifiers for text categorization. Technical report, Rutgers University, 2003.

Mario Figueiredo. Adaptive sparseness for supervised learning. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 25(9):1150–1159, 2003.

Mario A. T. Figueiredo. Adaptive sparseness using Jeffreys prior. In *Advances in Neural Information Processing Systems*, 2001.

Amir Globerson, Terry Y. Koo, Xavier Carreras, and Michael Collins. Exponentiated gradient algorithms for log-linear structured prediction. In *International Conference on Machine Learning*, 2007.

Tommi Jaakkola, Marina Meila, and Tony Jebara. Maximum entropy discrimination. In *Advances in Neural Information Processing Systems*, 1999.

Tony Jebara. Multitask sparsity via maximum entropy discrimination. *To appear in Journal of Machine Learning Research*, 2009.

Tony Jebara and Tommi Jaakkola. Feature selection and dualities in maximum entropy discrimination. In *Uncertainty in Artificial Intelligence*, 2000.

Ata Kaban. On Bayesian classification with laplace priors. *Pattern Recognition Letters*, 28(10): 1271–1282, 2007.

John Lafferty, Andrew McCallum, and Fernando Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *International Conference on Machine Learning*, 2001.

John Langford and John Shawe-Taylor. Pac-Bayes & margins. In *Advances in Neural Information Processing Systems*, 2003.

John Langford, Matthias Seeger, and Nimrod Megiddo. An improved predictive accuracy bound for averaging classifiers. In *International Conference on Machine Learning*, 2001.

Guy Lebanon and John Lafferty. Boosting and maximum likelihood for exponential models. In *Advances in Neural Information Processing Systems*, 2001.

Su-In Lee, Varun Ganapathi, and Daphne Koller. Efficient structure learning of Markov networks using l_1-regularization. In *Advances in Neural Information Processing Systems*, 2006.

Dong C. Liu and Jorge Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, (45):503–528, 1989.

David McAllester. Generalization bounds and consistency for structured labeling. In *Predicting Structured Data, edited by G. Bakir, T. Hofmann, B. Scholkopf, A. Smola, B. Taskar, and S. V. N. Vishwanathan. MIT Press*, 2007.

David McAllester. PAC-Bayesian model averaging. In *the Twelfth Annual Conference on Computational Learning Theory*, 1999.

Yuan (Alan) Qi, Martin Szummer, and Thomas P. Minka. Bayesian conditional random fields. In *International Conference on Artificial Intelligence and Statistics*, 2005.

Ariadna Quattoni, Michael Collins, and Trevor Darrell. Conditional random fields for object recognition. In *Advances in Neural Information Processing Systems*, 2004.

Lawrence R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, (77(2)):257–286, 1989.

Nathan D. Ratliff, J. Andrew Bagnell, and Martin A. Zinkevich. (online) subgradient methods for structured prediction. In *International Conference on Artificial Intelligence and Statistics*, 2007.

Robert E. Schapire, Yoav Freund, Peter Bartlett, and Wee Sun Lee. Boosting the margin: A new explanation for the effectiveness of voting methods. *The Annals of Statistics*, 26(5):1651–1686, 1998.

Peter Sollich. Bayesian methods for support vector machines: Evidence and predictive class probabilities. *Journal of Machine Learning Research*, (3):21–52, 2002.

Ben Taskar, Carlos Guestrin, and Daphne Koller. Max-margin Markov networks. In *Advances in Neural Information Processing Systems*, 2003.

Ben Taskar, Simon Lacoste-Julien, and Michael I. Jordan. Structured prediction via the extragradient method. In *Advances in Neural Information Processing Systems*, 2006.

Robert Tibshirani. Regression shrinkage and selection via the LASSO. *Journal of the Royal Statistical Society*, B(58):267–288, 1996.

Michael E. Tipping. Sparse bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, (1):211–244, 2001.

Ioannis Tsochantaridis, Thomas Hofmann, Thorsten Joachims, and Yasemin Altun. Support vector machine learning for interdependent and structured output spaces. In *International Conference on Machine Learning*, 2004.

Martin J. Wainwright, Pradeep Ravikumar, and John Lafferty. High-dimensional graphical model selection using l_1-regularized logistic regression. In *Advances in Neural Information Processing Systems*, 2006.

David Wipf, Jason Palmer, and Bhaskar Rao. Perspectives on sparse bayesian learning. In *Advances in Neural Information Processing Systems*, 2003.

Linli Xu, Dana Wilkinson, Finnegan Southey, and Dale Schuurmans. Discriminative unsupervised learning of structured predictors. In *International Conference on Machine Learning*, 2006.

Zhihua Zhang and Michael I. Jordan. Bayesian multicategory support vector machines. In *Uncertainty in Artificial Intelligence*, 2006.

Ji Zhu, Saharon Rosset, Trevor Hastie, and Rob Tibshirani. 1-norm support vector machines. In *Advances in Neural Information Processing Systems*, 2004.

Jun Zhu and Eric P. Xing. On primal and dual sparsity of Markov networks. In *International Conference on Machine Learning*, 2009.

Jun Zhu, Zaiqing Nie, Bo Zhang, and Ji-Rong Wen. Dynamic hierarchical Markov random fields for integrated web data extraction. *Journal of Machine Learning Research*, (9):1583–1614, 2008a.

Jun Zhu, Eric P. Xing, and Bo Zhang. Laplace maximum margin Markov networks. In *International Conference on Machine Learning*, 2008b.

Jun Zhu, Eric P. Xing, and Bo Zhang. Partially observed maximum entropy discrimination Markov networks. In *Advances in Neural Information Processing Systems*, 2008c.

Jun Zhu, Amr Ahmed, and Eric P. Xing. MedLDA: Maximum margin supervised topic models for regression and classification. In *International Conference on Machine Learning*, 2009a.

Jun Zhu, Eric P. Xing, and Bo Zhang. Primal sparse max-margin Markov networks. In *International Conference on Knowledge Discovery and Data Mining (KDD)*, 2009b.