Marginal Structured SVM with Hidden Variables

Wei Ping WPING@ICS.UCI.EDU
Qiang Liu QLIU1@ICS.UCI.EDU
Alexander Ihler IHLER@ICS.UCI.EDU
Department of Computer Sciences, UC Irvine

Abstract

In this work, we propose the marginal structured SVM (MSSVM) for structured prediction with hidden variables. MSSVM properly accounts for the uncertainty of hidden variables, and can significantly outperform the previously proposed latent structured SVM (LSSVM; Yu & Joachims (2009)) and other state-of-art methods, especially when that uncertainty is large. Our method also results in a smoother objective function, making gradient-based optimization of MSSVMs converge significantly faster than for LSSVMs. We also show that our method consistently outperforms hidden conditional random fields (HCRFs; Quattoni et al. (2007)) on both simulated and real-world datasets. Furthermore, we propose a unified framework that includes both our and several other existing methods as special cases, and provides insights into the comparison of different models in practice.

1. Introduction

Conditional random fields (CRFs) (Lafferty et al., 2001) and structured SVMs (SSVMs) (Taskar et al., 2003; Tsochantaridis et al., 2005) are standard tools for structured prediction in many important domains, such as computer vision (Nowozin & Lampert, 2011), natural language processing (Getoor & Taskar, 2007) and computational biology (e.g., Li et al., 2007). However, many practical cases are not well handled by these tools, due to the presence of latent variables or partially labeled datasets. For example, one approach to image segmentation classifies each pixel into a predefined semantic category. While it is expensive to collect labels for every single pixel (perhaps even impossible for ambiguous regions), partially labeled data are relatively easy to obtain (e.g., Verbeek & Triggs, 2007). Examples also arise in natural language processing, such as semantic role labeling, where the semantic predictions are inherently coupled with latent syntactic relations (Naradowsky et al., 2012). However, accurate syntactic annotations are unavailable in many language resources.

In past few years, several solutions have been proposed to address hidden variable problems in structured prediction. Perhaps the most notable of these are hidden conditional random fields (HCRFs) (Quattoni et al., 2007) and latent structured SVMs (LSSVMs) (Yu & Joachims, 2009), which are derived from conditional random fields and structured SVMs, respectively. However, both approaches have several shortcomings. CRF-based models often perform worse than SSVM-based methods in practical datasets, especially when the number of training instances is small or the model assumptions are heavily violated (e.g., Taskar et al., 2003). On the other hand, LSSVM relies on a joint maximum *a posteriori* (MAP) procedure that assigns the hidden variables to deterministic values, and does not take into account their uncertainty. Unfortunately, this can produce poor predictions of the output variables even for exact models (Liu & Ihler, 2013). A better approach is to average over possible states, corresponding to a *marginal MAP* inference task (Koller & Friedman, 2009; Liu & Ihler, 2013) that marginalizes the hidden variables before optimizing over the output variables.

Contributions. We propose a novel structured SVM algorithm that takes into account the uncertainty of the hidden variables, by incorporating marginal MAP inference that "averages" over the possible hidden states. We show that our method performs significantly better than LSSVM and other state of art methods, especially when the uncertainty of the hidden variables is high. Our method also inherits the general

Proceedings of the 31st International Conference on Machine Learning, Beijing, China, 2014. JMLR: W&CP volume 32. Copyright 2014 by the author(s).

advantages of structured SVMs and consistently outperforms HCRFs, especially when the training sample size is small. We also study the effect of different training algorithms under various models. In particular we show that gradient-based algorithms for our framework are much more efficient than for LSSVM, because our objective function is smoother than that of LSSVM as it marginalizes, instead of maximizes, over the hidden variables. Finally, we propose a unified framework that includes both our and existing methods as special cases, and provide general insights on the choice of models and optimization algorithms for practitioners.

We organize the rest of the paper as follows. In Section 2, we introduce related work. We present background and notation in Section 3, and derive our marginal structured SVM in Section 4. The unified framework is proposed in Section 5. Learning and inference algorithms for the model are presented in Section 6. We report experimental results in Section 7 and conclude the paper in Section 8.

2. Related Work

HCRFs naturally extend CRFs to include hidden variables, and have found numerous applications in areas such as object recognition (Quattoni et al., 2004) and gesture recognition (Wang et al., 2006). HCRFs have the same pros and cons as general CRFs; in particular, they perform well when the model assumptions hold and when there are enough training instances, but may otherwise perform badly. Alternatively, the LSSVM (Yu & Joachims, 2009) is an extension of structured SVM that handles hidden variables, with wide application in areas like object detection (Zhu et al., 2010) and human action recognition (Wang & Mori, 2009). However, LSSVM relies on a joint MAP procedure, and may not perform well when a non-trivial uncertainty exists in the hidden variables. Recently, Schwing et al. (2012) proposed an ϵ-extension framework for discriminative graphical models with hidden variables that includes both HCRFs and LSSVM as special cases.

A few recent works also incorporate uncertainty over hidden variables explicitly into their optimization frameworks. For example, Miller et al. (2012) proposed a max margin min-entropy (M3E) model that minimizes an uncertainty measure on hidden variables while performing max-margin learning. They assume that minimizing hidden uncertainty will improve the output accuracy. This is valid in some applications, such as object detection, where reducing the uncertainty of object location can improve the category prediction. However, in cases like image segmentation, the missing labels may come from ambiguous regions, and maintaining that ambiguity can be important. In another work, Kumar et al. (2012) proposes a learning procedure that encourages agreement between two separate models – one for predicting outputs and another for representing the uncertainty over the hidden variables. They model the uncertainty of hidden variable during training, and rely on a joint MAP procedure during prediction.

Our proposed method builds on recent work for marginal MAP inference (Koller & Friedman, 2009; Liu & Ihler, 2013), which averages over the hidden variables (or variables that are not of direct interest), and then optimizes over the output variables (or variables of direct interest). In many domains, marginal MAP can provide significant improvement over joint MAP estimation, which jointly optimizes hidden and output variables; recent examples include blind deconvolution in computer vision (Fergus et al., 2006; Levin et al., 2011) and relation extraction and semantic role labeling in natural language processing (Naradowsky et al., 2012). Unfortunately, marginal MAP tasks on graphical models are notoriously difficult; marginal MAP can be NP-hard even when the underlying graphical model is tree-structured (Koller & Friedman, 2009). Recently, Liu & Ihler (2013) proposed efficient variational algorithms that approximately solve marginal MAP. In our work, we use their mixed-product belief propagation algorithm as our inference component.

Sub-gradient decent (SGD) (Ratliff et al., 2007) and the concave-convex procedure (CCCP)(Yuille & Rangarajan, 2003) are two popular training algorithms for structured prediction problems. Generally, SGD is straightforward to implement and effective in practice, but may be slow to converge, especially on nonconvex and non-smooth objective functions as arise in LSSVMs. CCCP is a general framework for minimizing non-convex functions by transforming the nonconvex optimization into a sequence of convex optimizations by iteratively linearizing the non-convex component of the objective. It has been applied widely in many areas of machine learning, particularly when hidden variables or missing data are involved. We explore both these training methods and compare them across the various models we consider.

3. Structured Prediction with Hidden Variables

In this section we review the background on structured prediction with hidden variables. Assume we have structured input-output pairs $(x, y) \in \mathcal{X} \times \mathcal{Y}$, where $\mathcal{X}$, $\mathcal{Y}$ are the spaces of the input and output variables. In many applications, this input-output relationship is not only characterized by (x, y), but also depends on some unobserved hidden variables $h \in \mathcal{H}$. Suppose (x, y, h) follows a conditional model,

$$p(y, h|x; w) = \frac{1}{Z(x; w)} \exp\,[w^T \phi(x, y, h)], \qquad (1)$$

where $\phi(x, y, h) : \mathcal{X} \times \mathcal{Y} \times \mathcal{H} \to \mathbb{R}^D$ is a set of features that describe the relationships among the (x, y, h), and w are the corresponding weights. $Z(x; w)$ is the normalization constant, or *partition function*,

$$Z(x; w) = \sum_{y,h} \exp\,[w^T \phi(x, y, h)].$$

Assuming the weights w are known, the LSSVM of Yu & Joachims (2009) decodes the output variables given input variables x by a joint maximum *a posteriori* (MAP) inference,

$$[\tilde{y}(w), \tilde{h}(w)] = \arg\max_{y,h} p(y, h|x) = \arg\max_{y,h}\ w^T \phi(x, y, h).$$

This gives the optimal prediction of the (y, h)-pair, and one obtains a prediction on y by discarding the h component. Unfortunately, the optimal prediction for (y, h) jointly does not necessarily give an optimal prediction on y; instead, it may introduce strong biases even for simple cases (e.g., see Example 1 in Liu & Ihler (2013)). Intuitively, the joint MAP prediction is "overly optimistic", since it deterministically assign the hidden variables to their most likely states; this approach is not robust to the inherent uncertainty in h, which may cause problems if that uncertainty is significant.

To address this issue, we use marginal MAP predictor,

$$\begin{aligned} \hat{y}(w) &= \arg\max_{y \in \mathcal{Y}} \sum_h p(y, h|x; w) \\ &= \arg\max_{y \in \mathcal{Y}} \log \sum_h \exp\,[w^T \phi(x, y, h)], \end{aligned} \qquad (2)$$

which explicitly takes into account the uncertainty of the hidden variables. It should be noted that $\hat{y}(w)$ is in fact the Bayes optimal prediction of y, measured by zero-one loss. The main contribution of this work is to introduce a novel structured SVM-based method for training the marginal MAP predictor, which significantly improves over previous methods.

4. Marginal Structured SVM

In this section we derive our main algorithm, the marginal structured SVM (MSSVM), which minimizes an upper bound of the empirical risk function. Assume we have a set of training instances $S = \{(x_1, y_1), \cdots, (x_n, y_n)\} \in (\mathcal{X} \times \mathcal{Y})^n$. The risk is measured by an user-specified empirical loss function $\Delta(y_i, \hat{y}_i)$, which quantifies the difference between an estimator $\hat{y}_i$ and the correct output y_i. It is usually difficult to exactly minimize the loss function because it is typically non-convex and discontinuous (e.g., Hamming loss). Instead, one adopts surrogate upper bounds to overcome this difficulty.

Assume $\hat{y}_i(w)$ is the marginal MAP prediction on instance x_i as defined in (2). We upper bound the loss function $\Delta(y_i, \hat{y}_i(w))$ as,

$$\begin{aligned} &\Delta(y_i, \hat{y}_i(w)) \\ &\quad \le \Delta(y_i, \hat{y}_i(w)) + \log \sum_h \exp[w^T \phi(x_i, \hat{y}_i(w)), h)] \\ &\qquad - \log \sum_h \exp[w^T \phi(x_i, y_i, h)] \\ &\quad \le \max_y \Big\{ \Delta(y_i, y) + \log \sum_h \exp\,[w^T \phi(x_i, y, h)] \Big\} \\ &\qquad - \log \sum_h \exp\,[w^T \phi(x_i, y_i, h)], \end{aligned}$$

where the first inequality holds because $\hat{y}_i(w)$ is the marginal MAP prediction (2), and the second because it jointly maxmizes two terms.

Minimizing this upper bound over the training set with a L_2 regularization, we obtain the following objective function for our marginal structured SVM,

$$\begin{aligned} &\frac{1}{2}\|w\|^2 + C \sum_{i=1}^n \max_y \Big\{ \Delta(y_i, y) + \log \sum_h \exp[w^T \phi(x_i, y, h)] \Big\} \\ &\quad - C \sum_{i=1}^n \log \sum_h \exp\Big[w^T \phi(x_i, y_i, h)\Big]. \end{aligned} \qquad (3)$$

The constraint form of (3) can be found in the supplement. Note that the first part of the objective requires a loss-augmented *marginal MAP* inference, which marginalizes the hidden variables h and then optimizes over the output variables y, while the second part only requires a marginalization over the hidden variables. Both these terms and their gradients are intractable to compute on loopy graphical models, but can be efficiently approximated by mixed-product belief propagation (Liu & Ihler, 2013) and sum-product belief propagation (Wainwright & Jordan, 2008), respectively. We will discuss training algorithms for optimizing this objective in Section 6.

5. A Unified Framework

In this section, we compare our framework with a spectrum of existing methods, and introduce a more general framework that includes all these methods as special cases. To start, note that the objective function of the LSSVM (Yu & Joachims, 2009) is

$$\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\max_y \max_h \Big\{\Delta(y_i, y) + w^T\phi(x_i, y, h)\Big\} - C\sum_{i=1}^{n}\max_h \Big[w^T\phi(x_i, y_i, h)\Big]. \qquad (4)$$

Our objective in (3) is similar to (4), except replacing the *max* operator of h with the log-sum-exp function, the so called *soft-max* operator. One may introduce a "temperature" parameter that smooths between *max* and *soft-max*, which motivates a more general objective function that includes MSSVM, LSSVM and other previous methods as special cases,

$$\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\epsilon_y \log\sum_y \exp\ \Big\{\frac{1}{\epsilon_y}\Big[\Delta(y_i, y) + \ \epsilon_h \log\sum_h \exp\Big(\frac{w^T\phi(x_i, y, h)}{\epsilon_h}\Big)\Big]\Big\} - C\sum_{i=1}^{n}\epsilon_h \log\sum_h \exp\Big(\frac{w^T\phi(x_i, y_i, h)}{\epsilon_h}\Big), \qquad (5)$$

where ϵ_y and ϵ_h are temperature parameters that control how much uncertainty we want account for in y and h, respectively. Similar temperature-based approaches have been used both in structured prediction (Hazan & Urtasun, 2010; Schwing et al., 2012) and in other problems, such as semi-supervised learning (Samdani et al., 2012; Dhillon et al., 2012).

One can show (Lemma 1 in supplement) that objective (5) is an upper bound of the empirical loss function $\Delta(y_i, \hat{y_i}^{\epsilon_h}(w))$ over the training set, where the prediction $\hat{y_i}^{\epsilon_h}(w)$ is decoded by "annealed" marginal MAP,

$$\hat{y_i}^{\epsilon_h}(w) = \arg\max_y \log\sum_h \exp\Big[\frac{w^T\phi(x_i, y, h)}{\epsilon_h}\Big].$$

This framework includes a number of existing methods as special cases. It reduces to MSSVM in (3) if $\epsilon_y \to 0^+$ and $\epsilon_h = 1$, and LSSVM in (4) if $\epsilon_y \to 0^+$ and $\epsilon_h \to 0^+$. If we set $\epsilon_y = \epsilon_h = 1$, we obtain the loss-augmented likelihood objective in Volkovs et al. (2011), and further reduces to the standard likelihood objective of HCRFs if we assume $\Delta(y_i, y) \equiv 0$. Our framework also generalizes the ϵ-extension model by Schwing et al. (2012), which corresponds to the restriction that $\epsilon_y = \epsilon_h$. See Table 1 for a summarization of these model comparisons. In the sequel, we provide some general insights on selecting among these different models through our empirical evaluations.

Table 1. Model comparisons within our unified framework.

Model	$\epsilon_h \to 0^+(\max_h)$	$\epsilon_h = 1$ $(\sum_h)$
$\epsilon_y \to 0^+$ $(\max_y)$	LSSVM	MSSVM
$\epsilon_y = 1$ $(\sum_y)$	N/A	HCRFs
$\epsilon_y = \epsilon_h \in (0, 1)$	ϵ-extension model	

6. Training Algorithms

In this section, we introduce two optimization algorithms for minimizing the objective function in (3), including a sub-gradient descent (SGD) algorithm and a concave-convex procedure (CCCP). An empirical comparison of these two algorithms is given in the experiments of Section 7.

6.1. Sub-gradient Descent (SGD)

According to Danskin's theorem, the sub-gradient of MSSVM (3) is:

$$\nabla_w M = w + C\sum_{i=1}^{n}\mathbb{E}_{p(h|x_i,\hat{y}_i)}[\phi(x_i, \hat{y}_i, h)] - C\sum_{i=1}^{n}\mathbb{E}_{p(h|x_i,y_i)}[\phi(x_i, y_i, h)], \qquad (6)$$

where,

$$\hat{y}_i = \arg\max_{y\in\mathcal{Y}}\Big\{\Delta(y_i, y) + \log\sum_h \exp[w^T\phi(x_i, y, h)]\Big\} \qquad (7)$$

is the loss-augmented marginal MAP prediction, which can be approximated via mixed-product belief propagation as described in Liu & Ihler (2013). The $\mathbb{E}_{p(h|x_i,\hat{y}_i)}$ and $\mathbb{E}_{p(h|x_i,y_i)}$ denote the expectation over the distributions $p(h|x_i, \hat{y}_i)$ and $p(h|x_i, y_i)$, respectively. Both expectations can similarly be approximated using the marginal probabilities obtained from belief propagation. See Algorithm 1 for the sub-gradient descent (SGD) algorithm for MSSVM.

Furthermore, one can show (Lemma 2 in supplement) that the (sub-)gradient of the unified framework (5) is

$$\nabla_w U = w + C\sum_{i=1}^{n}\mathbb{E}_{p^{(\epsilon_y,\epsilon_h)}(y,h|x_i)}[\phi(x_i, y, h)] - C\sum_{i=1}^{n}\mathbb{E}_{p^{\epsilon_h}(h|x_i,y_i)}[\phi(x_i, y_i, h)]. \qquad (8)$$

where the corresponding temperature controlled distribution is defined as,

$$p^{\epsilon_h}(h|x_i, y) \propto \exp\Big[\frac{w^T\phi(x_i, y, h)}{\epsilon_h}\Big],$$

Algorithm 1 Sub-gradient Descent for MSSVM

```
Input: number of iterations $T$, learning rate $\eta$
Output: the learned weight vector $w^*$
$w^0 = 0$
for $t = 1$ to $T$ do
    $\nabla_w = w^{t-1}$
    for $i = 1$ to $n$ do
        1. Calculate $\phi_m = \mathbb{E}_{p(h|x_i,\hat{y}_i)}[\phi(x_i, \hat{y}_i, h)]$ by
        mixed-product BP ($\hat{y}_i$ is defined in (7))
        2. Calculate $\phi_s = \mathbb{E}_{p(h|x_i,y_i)}[\phi(x_i, y_i, h)]$ by
        sum-product BP
        3. $\nabla_w \leftarrow \nabla_w + C \cdot (\phi_m - \phi_s)$
    end for
    $w^t \leftarrow w^{t-1} - \eta\nabla_w$
end for
$w^* \leftarrow w^t$
```

$$p^{(\epsilon_y,\epsilon_h)}(y|x_i) \propto \exp\Big\{\frac{1}{\epsilon_y}\big[\Delta(y, y_i) + \epsilon_h \log \sum_h \exp\big(\frac{w^T\phi(x_i, y, h)}{\epsilon_h}\big)\big]\Big\},$$
$$p^{(\epsilon_y,\epsilon_h)}(y, h|x_i) = p^{\epsilon_h}(h|x_i, y) \cdot p^{(\epsilon_y,\epsilon_h)}(y|x_i).$$

Exactly as in Table 1, this reduces to the sub-gradient of MSSVM (6) if $\epsilon_y \to 0^+$ and $\epsilon_h = 1$, the sub-gradient of LSSVM if $\epsilon_y \to 0^+$ and $\epsilon_h \to 0^+$, and the gradient of HCRF if $\epsilon_y = 1$, $\epsilon_h = 1$ and $\Delta(y, y_i) \equiv 0$. One can simply substitute these (sub-)gradients into Algorithm 1 to obtain the corresponding training algorithms of LSSVM and HCRF. In those cases, max-product BP and sum-product BP can be used to approximate the inference operations instead.

6.2. CCCP Training Algorithm

The concave-convex procedure (CCCP) (Yuille & Rangarajan, 2003) is a general non-convex optimization algorithm with wide application in machine learning. It is based on the idea of rewriting the non-convex objective function into a sum of a convex function and a concave function (or equivalently a difference of two convex functions), and transforming the non-convex optimization problem into a sequence of convex sub-problems by linearizing the concave part. CCCP provides a straightforward solution for our problem, since the objective functions of all the methods we have discussed – in (3), (4) and (5) – are naturally differences of two convex functions. For example, the MSSVM objective in (3) can be naturally written as,

$$f(w) = f^+(w) - f^-(w),$$

Algorithm 2 CCCP Training of MSSVM

```
Input: number of outer iterations $T$, learning rate
$\eta$, tolerance $\epsilon$ for inner loops
Output: the learned weight vector $w^*$
for $t = 1$ to $T$ do
    $u^t = 0$
    for $i = 1$ to $n$ do
        1. Calculate $\phi_s = \mathbb{E}_{p(h|x_i,y_i)}[\phi(x_i, y_i, h)]$ by
           sum-product BP
        2. $u^t = u^t + \phi_s$
    end for
    $w^t = w^{t-1}$
    repeat
        $\nabla_w = w_t$
        for $i = 1$ to $n$ do
            1. Calculate $\phi_m = \mathbb{E}_{p(h|x_i,\hat{y}_i)}[\phi(x_i, \hat{y}_i, h)]$ by
               mixed-product BP ($\hat{y}_i$ is defined in (7))
            2. $\nabla_w \leftarrow \nabla_w + C\phi_m - Cu_t$
        end for
        $w^t \leftarrow w^t - \eta\nabla_w$
    until $||\nabla_w|| \leq \epsilon$
end for
$w^* \leftarrow w^t$
```

$$\text{where} \quad f^+(w) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^n \max_y \Big\{\Delta(y_i, y) + \log\sum_h \exp[w^T\phi(x_i, y, h)]\Big\},$$
$$f^-(w) = C\sum_{i=1}^n \log\sum_h \exp[w^T\phi(x_i, y_i, h)].$$

Denoting the parameter vector at iteration t by w^t, the CCCP algorithm updates to new parameters w^{t+1} by minimizing a convex surrogate function where $f^-(w)$ is linearized:

$$w^{t+1} \leftarrow \arg\min_w \{f^+(w) - w^T\nabla f^-(w^t)\},$$
$$\text{where} \quad \nabla f^-(w^t) = C\sum_i \mathbb{E}_{p(h|x_i,y_i)}[\psi(x_i, y_i, h)]$$

is the gradient of $f^-(w)$ at w_t and its expectation can be evaluated (approximately) by belief propagation. See Algorithm 2 for more details of CCCP for the MSSVM.

7. Experiments

In this section, we compare our MSSVM with other state-of-art methods on both simulated and real-world datasets. We demonstrate that the MSSVM significantly outperforms LSSVM, max-margin min-entropy (M3E) model (Miller et al., 2012) and loss-based

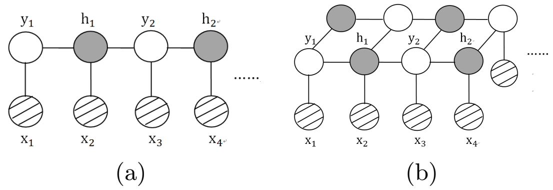


Figure 1. (a) The hidden chain (b) 2D grid model used in our simulation experiment. The shaded nodes denote hidden variables h, while the unshaded nodes are the output variables y and nodes with hatching are the input x.

learning by modeling latent variable(ModLat) (Kumar et al., 2012), especially when the uncertainty over hidden variables is high. Our method also largely outperforms HCRFs in all experiments, especially with a small training sample size.

7.1. Simulated Data

We simulate both training and testing data from a pairwise Markov random field (MRF) over graph $G = (V, E)$ with discrete random variables taking values in $\{0, 1, 2, 3\}^n$, given by

$$p(x, y, h|w) \propto$$
$$\exp\Big[\sum_{x_i \in V} w_{x_i}^T \phi(x_i) + \sum_{y_j \in V} w_{y_j}^T \phi(y_j) + \sum_{h_k \in V} w_{h_k}^T \phi(h_k)$$
$$+ \sum_{(x_i,y_j)\in E} w_{(x_i,y_j)}^T \phi(x_i, y_j) + \sum_{(x_i,h_k)\in E} w_{(x_i,h_k)}^T \phi(x_i, h_k)$$
$$+ \sum_{(y_j,h_k)\in E} w_{(y_j,h_k)}^T \phi(y_j, h_k)\Big], \qquad (9)$$

where the graph structure G is either a "hidden chain" (40 nodes) or a 2D grid (size $6 \times 6 \times 2$), as illustrated in Figure 1. The log-linear weights w are randomly generated from normal distributions. The singleton parameters w_{x_i}, w_{y_j} and w_{h_k} are drawn from $N(0,\ \sigma_x^2 \cdot I)$, $N(0,\ \sigma_y^2 \cdot I)$ and $N(0, \sigma_h^2 \cdot I)$, respectively, corresponding to indicator vectors $\phi(x_i)$, $\phi(y_j)$ and $\phi(h_k)$. The pairwise parameters $w_{(y_j,h_k)=(s,t)}$, $w_{(x_i,y_j)=(r,s)}$ and $w_{(x_i,h_k)=(r,t)}$ are drawn from $N(0, \sigma_{yh}^2)$, $N(0, \sigma_{xy}^2)$ and $N(0, \sigma_{xh}^2)$, respectively, corresponding to indicators $\phi(y_j = s,\ h_k = t)$, $\phi(x_i = r,\ y_j = s)$ and $\phi(x_i = r,\ h_k = t)$. Note that the variance parameters σ_h and σ_{yh} control the degree of uncertainty in the hidden variables and their importance for estimating the output variables y: the uncertainty of h is high for small values of σ_h, and the correlation between h and y is high when σ_{yh} is large. We sample 20 training instances and 100 test instances from both the hidden chain MRF and 2D grid MRF. We set $\sigma_x = \sigma_y = \sigma_h = 0.1$, $\sigma_{yh} = \sigma_{yx} = \sigma_{hx} = 2$. Then, we train our MSSVM, LSSVM and HCRF models using both SGD and CCCP. Hamming loss is used in both training and evaluation. In our experiments, we always set the regularization weight $C = 1$. See Table 2 for the results across different algorithms. We can see that our MSSVM always achieves the highest accuracy when using either training algorithm. It is worth noting that LSSVM obtains a significantly better result using CCCP than SGD; this is mainly due to SGD's difficulty converging on the piecewise linear objective of LSSVM.

Table 2. Average accuracy (%) of MSSVM, LSSVM, HCRFs using SGD and CCCP when the data are simulated from 40-node hidden chain and 6×6 2D-grid graph as shown in Figure 1. The results are averaged over 20 random trails.

Hidden Chain	MSSVM	LSSVM	HCRFs
SGD	**69.20**	66.87	68.75
CCCP	**69.63**	67.91	69.03

2D-grid graph	MSSVM	LSSVM	HCRFs
SGD	**74.12**	71.96	73.51
CCCP	**74.08**	73.38	73.62

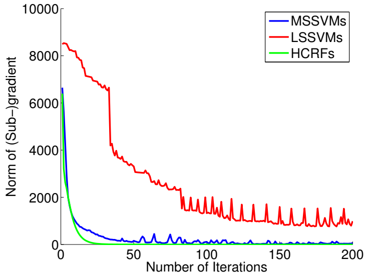


Figure 2. Convergence behavior of (sub-)gradient descent on MSSVMs, LSSVMs and HCRFs.

Empirical Convergence of SGD and CCCP. Using sub-gradient descent with learning rate $\eta_M = 0.02$, we found that for our MSSVM, training error converged quickly (within 50 iterations). However, sub-gradient descent on the LSSVM would only converge using a much smaller learning rate ($\eta_L = 0.001$), and converged more slowly (usually after 250 iterations). This effect is mainly because the LSSVM hard-max makes the objective function nonsmooth, causing sub-gradient descent to be slow to converge. On the other hand, gradient descent on HCRFs converges more easily and quickly than either MSSVM or LSSVM, because its objective function is smoother. Figure 2 shows the oscillation during the iteration of (sub-)gradient descent for each model, and empirically illustrates the convergence process.

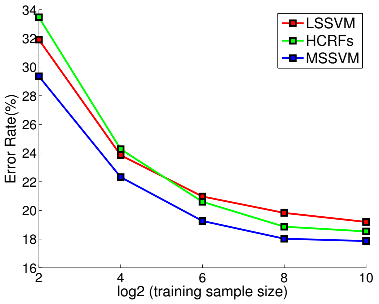


Figure 3. The error rate of MSSVM, LSSVM and HCRFs as the training sample size increases. Results are averaged over 5 random trials.

Table 3. The accuracy (%) of MSSVM, LSSVM, HCRFs, M3E and ModLat under different σ_h, which governs the level of uncertainty in the hidden variables. Small values of σ_h correspond to high uncertainty in hidden variables. Results are averaged over 20 random trials.

σ_h	MSSVM	LSSVM	HCRFs	M3E	ModLat
10	79.30	**79.46**	78.68	79.04	77.16
1	70.00	**70.07**	69.88	68.53	67.91
0.5	**67.24**	65.98	66.66	66.05	65.15
0.1	**69.63**	67.91	69.03	65.19	67.96
0.01	**73.88**	71.38	72.58	67.21	71.52
1e-3	**72.08**	69.24	70.88	65.48	66.54
Avg.	**72.02**	70.67	71.28	68.58	69.37

We also observe CCCP converging faster than SGD (using smaller number of inference steps), especially for LSSVM, since CCCP transforms the complex piecewise linear objective into a sequence of easier convex sub-problems. In our empirical study, CCCP always converged well even using approximate inference and non-convex objectives.[1] To provide a fair comparison, all methods are trained using the CCCP algorithm in the sequel.

Training Sample Size. We compared the influence of sample size for each method by ranging the training size from 2^2 to 2^{10} (with a testing size of 500). The data are all simulated from a MRF on the 20-node hidden chain shown in Figure 1(a). We set $\sigma_x = \sigma_y = \sigma_h = 0.1$ and $\sigma_{yh} = \sigma_{yx} = \sigma_{hx} = 2$ as before.

Results are shown in Figure 3. We found that our MSSVM always considerably outperforms LSSVM, and largely outperforms HCRFs when the training sample sizes are small. HCRFs perform worse than LSSVM for few training data, but outperform LSSVM as the training sample increases.

Our experiment shows that MSSVM consistently outperforms HCRFs even with reasonably large training sets on a relatively simple toy model. Although the maximum likelihood estimator (as used in HCRFs) is generally considered asymptotically optimal if the model assumptions are correct, this assumes a sufficiently large training size, which may be difficult to acheive in practice. Given enough data (and the correct model), the HCRF should thus eventually improve, but this seems unrealistic since most applications are likely to exhibit high dimensional parameters and relatively few training instances. Additional analysis of the test likelihood and prediction accuracy can be found in the supplement.

Uncertainty of Hidden Variables. We investigate the influence of uncertainty in the hidden variables for each method by adjusting the noise level σ_h, which controls the uncertainty of the hidden variables. We draw 20 training samples and 100 test samples from a MRF on a 40-node hidden chain shown in Figure 1(a), with fixed $\sigma_x = \sigma_y = 0.1$ and $\sigma_{yh} = \sigma_{yx} = \sigma_{hx} = 2$. For comparison, we also evaluate the performance of M3E (Miller et al., 2012) and ModLat (Kumar et al., 2012).

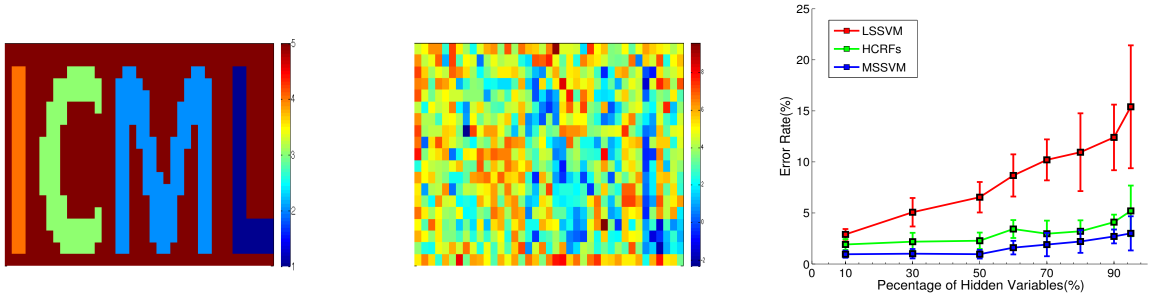
Results are shown in Table 3. We find that our MSSVM is competitive with LSSVM and M3E when the uncertainty in the hidden variables is low, and becomes significantly better than them as the uncertainty increases. Because LSSVM uses the joint MAP, it does not take into account this uncertainty. On the other hand, M3E explicitly tries to minimize this uncertainty, which can also mislead the prediction. Our MSSVM consistently outperforms HCRFs for moderate training sample sizes. Note that current implementations of M3E and ModLat are built on the Latent SVM$^{\text{struct}}$ package[2] which often fails in loopy models (see e.g. Figure 2 in Schwing et al. (2012)). Therefore, we only provide their results on chain models.

7.2. Image Segmentation

In this section, we evaluate our MSSVM method on the task of segmenting weakly labeled images. Our settings are motivated by the experiments in (Schwing et al., 2012). We assume a ground truth image of 20×40 pixels as shown in Figure 4(a), where each pixel i has a label y_i taking values in $\{1, \cdots, 5\}$. The observed image x is obtained by adding Gaussian noise,

[1] However, it is challenging to provide rigorous convergence guarantees for the non-convex & intractable setting, and not really the focus of this paper.

[2] http://www.cs.cornell.edu/~cnyu/latentssvm/



(a) (b) (c)

Figure 4. (a) The ground truth image. (b) An example of an observed noisy image. (c) The performance of each algorithm as the percentage of missing labels varies from 10% to 95%. Results are averaged over 5 random trials, each using 10 training instances and 10 test instances.

$N(0, 5)$, on the ground truth image. We use the 2D-grid model as in Figure 1 (b), with local features $\phi(y_i, x_i) = e_{y_i} \otimes x_i$ and pairwise features $\phi(y_i, y_j) = e_{y_i} \otimes e_{y_j} \in \mathbb{R}^{5\times5}$ as defined in Nowozin & Lampert (2011), where e_{y_i} is the unit normal vector with entry one on dimension y_i and $\otimes$ is the outer product. The set of missing labels (hidden variables) are determined at random, in proportions ranging from 10% to 95%. The performance of MSSVM, LSSVM, and HCRFs are evaluated using the CCCP algorithm.

Figure 4(c) lists the performance of each method as the percentage of missing labels is increased. We can see that the performance of LSSVM degrades significantly as the number of hidden variables grows. Most notably, MSSVM is consistently the best method across all settings. This can be explained by MSSVM combining both the max-margin property and the improved robustness given by properly accounting for uncertainty in the hidden labels.

7.3. Object Categorization

Finally, we evaluate our MSSVM method on the task of object categorization using partially labeled images. We use the Microsoft Research Cambridge data set (Winn et al., 2005), consisting of 240 images with 213×320 pixels and their partial pixel-level labelings. Modeled on the approach outlined in Verbeek & Triggs (2007), we use 20×20 pixel patches with centers at 10 pixel intervals and treat each patch as a node in our model. This results in a 20×31 grid model as in Figure 1 (b). The local features of each patch are encoded using texture and color descriptors. For texture, we compute the 128-dimensional SIFT descriptor of the patch and vector quantize it into a 500-word codebook, learned by k-means clustering of all patches in the entire dataset. For color, we take 48-dimensional RGB color histogram for each patch. In our experiment, we select the 5 most frequent categories in the dataset and use 2-fold cross validation for testing.

Table 4. Average patch level accuracy (%) of MSSVM, LSSVM, HCRFs for MSRC data by 2-fold cross validation

MSRC Data	MSSVM	LSSVM	HCRFs
Building	**72.4**	70.7	71.7
Grass	**89.7**	88.9	88.3
Sky	**88.3**	85.6	88.2
Tree	**71.9**	71.0	70.1
Car	**70.8**	69.4	70.2

Table 4 shows the accuracies of each method across the various categories. Again, we find that MSSVM consistently outperforms other methods across all categories, which can be explained by both the superiority of SSVM-based methods for moderate sample size and the improved robustness by accounting for uncertainty in the missing labels.

8. Conclusion

We proposed a novel structured SVM method for structured prediction with hidden variables. We demonstrate that our MSSVM consistently outperforms state-of-the-art methods in both simulated and real-world datasets, especially when the uncertainty of hidden variables is large. Compared to the popular LSSVM, our MSSVM is easy to optimize due to the smoothness of its objective function. We also provide a unified framework which includes our method as well as a spectrum of previous methods as special cases.

Acknowlegements This work was sponsored in part by NSF grants IIS-1065618 and IIS-1254071, AFRL, and DARPA. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of NSF, DARPA, AFRL, or the U.S. government.

References

Dhillon, P., Keerthi, S. Sathiya, Bellare, K., Chapelle, O., and and, S. Sundararajan. Deterministic annealing for semi-supervised structured output learning. In *Proceedings of AISTAT*, pp. 299–307, 2012.

Fergus, R., Singh, B., Hertzmann, A., Roweis, S. T., and Freeman, W. T. Removing camera shake from a single photograph. In *Proceeding of ACM SIGGRAPH*, pp. 787–794, 2006.

Getoor, L. and Taskar, B. *Introduction to statistical relational learning*. The MIT press, 2007.

Hazan, T. and Urtasun, R. A primal-dual message-passing algorithm for approximated large scale structured prediction. In *Proceedings of NIPS*, 2010.

Koller, D. and Friedman, N. *Probabilistic graphical models: principles and techniques*. The MIT press, 2009.

Kumar, P., Packer, B., and Koller, D. Modeling latent variable uncertainty for loss-based learning. In *Proceedings of ICML*, pp. 465–472, 2012.

Lafferty, J., McCallum, A., and Pereira, F. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of ICML*, pp. 282–289, 2001.

Levin, A., Weiss, Y., Durand, F., and Freeman, W.T. Efficient marginal likelihood optimization in blind deconvolution. In *Proceedings of CVPR*, pp. 2657–2664, 2011.

Li, M. H., Lin, L., Wang, X.L., and Liu, T. Protein–protein interaction site prediction based on conditional random field. *Bioinformatics*, 23, 2007.

Liu, Q. and Ihler, A. Variational algorithms for marginal map. *JMLR*, 14:3165–3200, 2013.

Miller, K., Kumar, P., Packer, B., Goodman, D., and Koller, D. Max-margin min-entropy models. In *Proceedings of AISTATS*, pp. 779–787, 2012.

Naradowsky, J., Riedel, S., and Smith, D. Improving NLP through marginalization of hidden syntactic structure. In *Proceeding of EMNLP*, pp. 810–820, 2012.

Nowozin, S. and Lampert, C. Structured prediction and learning in computer vision. *Foundations and Trends in Computer Graphics and Vision*, 6, 2011.

Quattoni, A., Collins, M., and Darrell, T. Conditional random fields for object recognition. In *Proceedings of NIPS*, pp. 1097–1104, 2004.

Quattoni, A., Wang, S., Morency, L., Collins, M., and Darrell, T. Hidden conditional random fields. *IEEE Transactions on PAMI*, 29:1848–1852, 2007.

Ratliff, N., Bagnell, J. A., and Zinkevich, M. (Online) Subgradient methods for structured prediction. In *Proceedings of AISTATS*, pp. 380–387, 2007.

Samdani, R., Chang, M.W., and Roth, D. Unified expectation maximization. In *Proceedings of NAACL*, pp. 688–698, 2012.

Schwing, A., Hazan, T., Pollefeys, M., and Urtasun, R. Efficient structured prediction with latent variables for general graphical models. In *Proceedings of ICML*, pp. 959–966, 2012.

Taskar, B., Guestrin, C., and Koller, D. Max-margin Markov networks. In *Proceedings of NIPS*, 2003.

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

Verbeek, J. and Triggs, B. Scene segmentation with CRFs learned from partially labeled images. In *Proceedings of NIPS*, pp. 1553–1560, 2007.

Volkovs, M., Larochelle, H., and Zemel, R. S. Loss-sensitive training of probabilistic conditional random fields. *Technical report*, 2011.

Wainwright, M. and Jordan, M.I. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1–2: 1–305, 2008.

Wang, S.B, Quattoni, A., Morency, L., and Demirdjian, D. Hidden conditional random fields for gesture recognition. In *Proceedings of CVPR*, pp. 1521–1527, 2006.

Wang, Y. and Mori, G. Max-margin hidden conditional random fields for human action recognition. In *Proceedings of CVPR*, pp. 872–879, 2009.

Winn, J., Criminisi, A., and Minka, T. Object categorization by learned universal visual dictionary. In *Proceedings of ICCV*, pp. 1800–1807, 2005.

Yu, C. and Joachims, T. Learning structural SVMs with latent variables. In *Proceedings of ICML*, pp. 1169–1176, 2009.

Yuille, A. L. and Rangarajan, A. The concave-convex procedure. *Neural Computation*, 15:915–936, 2003.

Zhu, L., Chen, Y., Yuille, A., and Freeman, W. Latent hierarchical structural learning for object detection. In *Proceedings of CVPR*, pp. 1062–1069, 2010.