Learning Convex QP Relaxations for Structured Prediction

Jeremy Jancsary JERMYJ@MICROSOFT.COM
Sebastian Nowozin SENOWOZI@MICROSOFT.COM
Carsten Rother CARROT@MICROSOFT.COM
Microsoft Research Cambridge, 21 Station Road, Cambridge, CB1 2FB, United Kingdom

Abstract

We introduce a new large margin approach to discriminative training of intractable discrete graphical models. Our approach builds on a convex quadratic programming relaxation of the MAP inference problem. The model parameters are trained directly within this restricted class of energy functions so as to optimize the predictions on the training data. We address the issue of how to parameterize the resulting model and point out its relation to existing approaches. The primary motivation behind our use of the QP relaxation is its computational efficiency; yet, empirically, its predictive accuracy compares favorably to more expensive approaches. This makes it an appealing choice for many practical tasks.

1. Introduction

Discriminative training of structured prediction models is a significant computational challenge that becomes even more difficult if exact inference in the model is intractable. Yet, this situation is not uncommon at all: For instance, even plain grid-structured graphical models, as commonly used in computer vision, can lead to NP-hard inference problems.

Discriminative learning formulations such as CRFs (Lafferty et al., 2001) or M3Ns (Taskar et al., 2003) require that an inference problem be solved repeatedly for each training example, at each step of an iterative solver. Consequently, approximate training objectives such as pseudolikelihood (Besag, 1975) or piecewise training (Sutton & McCallum, 2009) have commonly been used to avoid inference during training altogether. At test time, the prediction of the intractable model is then obtained approximately.

Proceedings of the 30th International Conference on Machine Learning, Atlanta, Georgia, USA, 2013. JMLR: W&CP volume 28.

More recently, several authors (Wainwright, 2006; Kulesza & Pereira, 2007; Finley & Joachims, 2008) have discussed the benefits of using the *same* approximate "inference machine" at test time *and* during training, to ensure compatibility. One well-understood approach towards this end relaxes the intractable MAP inference problem via a linear programming formulation due to Koval & Schlesinger (1976). Although there has been steady progress in solving this LP (e.g. Ravikumar et al., 2010; Savchynskyy et al., 2012), it is still recognized as a difficult practical problem. For instance, some of the most efficient specialized solvers (Kolmogorov, 2006; Globerson & Jaakkola, 2007) work on the dual of the LP relaxation and ensure monotonic descent, but do not guarantee global convergence.

Here, we consider a different relaxation of MAP inference and explore its utility in learning discriminative models. The *quadratic* programming relaxation (Ravikumar & Lafferty, 2006) has several advantages over its LP counterpart: Its optimization involves fewer variables and a separable constraint set (§3.2); dense interaction matrices can be incorporated if they allow for efficient multiplication (§5.4); and it leads to a differentiable large margin parameter estimation objective (§4.2). On the downside, it is strictly dominated by the LP relaxation in theory (Kumar et al., 2009), though the practical impact is mostly unclear. Moreover, tractability of the QP relaxation is typically ensured by convexifying the problem post-hoc (§3.2), so it is not immediately clear how to use it for learning.

We make the following contributions: a) We provide specific discriminative parameterizations that *directly* ensure convexity of the QP relaxation, avoiding the need for post-hoc "convexification" and enabling its use in large margin learning. b) We provide, to our knowledge, the first empirical study of discriminative training using the QP relaxation, comparing it directly to related approaches (including the LP relaxation) on four difficult structured prediction problems. Our results show that the QP relaxation is highly efficient and provides competitive predictive performance.

2. Related Work

We are not aware of previous work that uses convex QP relaxations in learning; however, several authors have considered training of discriminative models when exact inference is intractable. Kulesza & Pereira (2007) give technical conditions under which the use of LP-relaxed inference yields provably good results in standard learning frameworks. Finley & Joachims (2008) further show that large margin learning with relaxations discourages fractional predictions at test time. While only the LP relaxation is actually considered in their paper, the results apply equally to the convex QP relaxation. Moreover, building on the aforementioned work, Martins et al. (2009) derive even stronger guarantees for learning with relaxations.

A number of authors have attempted to accelerate approximate discriminative learning by dualizing the LP relaxation in the loss function, keeping the dual variables of all examples in memory. Meshi et al. (2010) alternate block coordinate updates on these dual variables and stochastic gradient descent on the model parameters. Hazan & Urtasun (2010) devise a similar approach; in addition, their model contains a one-parameter extension that interpolates between large margin learning and maximum conditional likelihood learning. Komodakis (2011) uses a dual formulation motivated by dual decomposition, and again updates model parameters and dual variables jointly. We do not follow the same strategy for the QP relaxation, but rather solve the inference subproblem exactly at each step, which is fast and requires less memory.

A different approach to attaining tractability, which has been followed for binary output variables, is to enforce submodularity of the inference subproblem (Taskar et al., 2006; Franc & Savchynskyy, 2008). The subproblem can then be solved efficiently using graph cuts, enabling exact training within this restricted family. Our approach is related in that the inference problem is restricted to belong to a particular class in which we can learn efficiently; but unlike graph cuts, our approach can handle multiple labels natively.

Finally, several authors have attempted to minimize the empirical risk of arbitrary "inference machines" mapping from input to a labeling by *directly* computing the loss on the prediction obtained from the model (Stoyanov et al., 2011; Domke, 2011). Among these methods, our approach is most closely related to the logistic random field (LRF) of Tappen et al. (2008). Unlike our approach, the logistic random field uses an *unconstrained* quadratic energy. Moreover, it is originally parameterized very restrictively and unfortunately leads to a non-convex learning problem.

3. Background

Consider an undirected graphical model defined over n nodes $i \in \mathcal{V}$, each of which can be in one of k states $y_i \in \{1, \dots, k\}$. The likelihood of a joint state $y = (y_i)_{i \in \mathcal{V}}$ is described in terms of an energy E,

$$p(y; \theta) \propto e^{-E(y;\theta)}, \tag{1}$$

which is assumed to decompose over edges $(i, j) \in \mathcal{E}$:

$$E(y; \theta) = \textstyle\sum_i \theta_i(y_i) + \sum_{(i,j)} \theta_{ij}(y_i, y_j). \tag{2}$$

For training and prediction, we want to obtain the joint state that minimizes the energy, referred to as as the min-sum problem or MAP estimation.

3.1. LP Formulation of the Min-Sum Problem

Observe that energy E can be defined equivalently using indicator vectors $\phi(y)$ selecting the appropriate components of the exponential parameters θ:

$$\underset{y \in \mathcal{Y}}{\text{minimize}} \quad \phi(y)'\theta. \tag{3}$$

The above optimization occurs over a finite number of discrete points. By standard results, it is equivalent to optimize over the convex hull $\mathcal{M} = \text{conv}\{\phi(y)\}_{y \in \mathcal{Y}}$,

$$\underset{\mu \in \mathcal{M}}{\text{minimize}} \quad \mu'\theta, \tag{4}$$

and any solution lies at one of the corner points of $\mathcal{M}$, known as the marginal polytope (Wainwright & Jordan, 2008). Due to the exponential nature of $\mathcal{M}$, only special cases of the above LP can be solved.

LP relaxation. Instead, one can always optimize over the local polytope $\mathcal{L}$, an approximation ensuring only local marginal consistency (Wainwright & Jordan, 2008). This can introduce fractional solutions, as shown in Fig. 1a. Moreover, the resulting LP has $O(k|\mathcal{V}| + k^2|\mathcal{E}|)$ variables and $O(2k|\mathcal{E}|)$ constraints. A dual formulation typically optimized by message passing algorithms (e.g. Kolmogorov, 2006) is unconstrained, but still involves $O(2k|\mathcal{E}|)$ variables, which can be prohibitively expensive for dense graphs.

3.2. QP Formulation of the Min-Sum Problem

In contrast, we will work with a different formulation of the min-sum problem involving only $O(k|\mathcal{V}|)$ variables and $O(k|\mathcal{V}|)$ separable constraints. In particular, by arranging all pairwise parameters θ_{ij} into a matrix Θ, and all θ_i into a vector θ, an alternative (but exact) definition of the min-sum problem is given by

$$\underset{y \in \mathcal{Y}}{\text{minimize}} \quad \psi(y)'\theta + \frac{1}{2}\psi(y)'\Theta\psi(y). \tag{5}$$

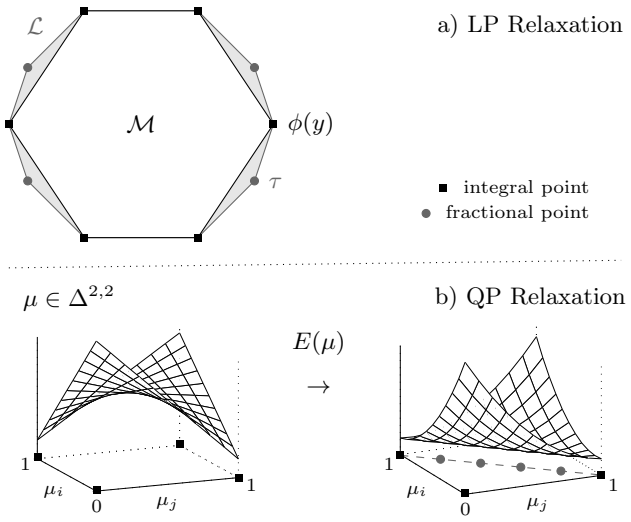


Figure 1. Tractable approximations of the MAP problem. In the exact LP and QP formulations, solutions are integral and occur at corner points of the constraint set. a) In the LP relaxation, corner points need no longer be integral. b) In the convex QP relaxation, convexification of the energy can lead to additional solutions in the interior.

Here, the convex hull of all indicator vectors $\psi(y)$,

$$\begin{aligned}\Delta^{n,k} &= \operatorname{conv}\{\psi(y)\}_{y\in\mathcal{Y}} \\ &= \{\mu \geq 0 \mid \textstyle\sum_{y_i} \mu_i(y_i) = 1\},\end{aligned} \tag{6}$$

consists of $n = |\mathcal{V}|$ unit simplices; one per variable. Each μ_i can be thought of as encoding the marginal probabilities of the k states of a single node.

QP relaxation. Similarly to the exact LP, it is known (Ravikumar & Lafferty, 2006) that optimizing the quadratic energy over this set,

$$E(\mu;\theta) = \mu'\theta + \frac{1}{2}\mu'\Theta\mu, \quad \mu \in \Delta^{n,k}, \tag{7}$$

yields corner points $\psi(y)$ encoding exact solutions y. However, since the energy is generally non-convex, direct minimization is still difficult.

Ravikumar & Lafferty (2006) propose to convexify the energy by adding a diagonal term D such that the matrix Θ becomes diagonally dominant, and to subtract the same term from the unary potentials,

$$\breve{\Theta} = \Theta + D \quad \text{and} \quad \breve{\theta} = \theta - \operatorname{vec}(D). \tag{8}$$

This approximation can be justified by the fact that under the indicator formulation (5), equivalence is maintained. However, the guarantee does not carry over to continuous energy (7): Indeed, fractional solutions can be introduced in the process (see Fig. 1b).

4. Learning Convex QP Relaxations

The QP relaxation is attractive due to its small number of variables and constraints. But convexifying the energy post-hoc, akin to Ravikumar & Lafferty (2006), is unsuitable for use in a learning framework. Instead, we aim at *directly* learning the best parameters within the class of convex quadratic energies.

4.1. Parameterization

In a discriminative model, the exponential parameters are a function of the observed input x and the model parameters w. Hence, our energy is of the form

$$E(\mu|x;w) = \mu'\theta(x;w) + \frac{1}{2}\mu'\Theta(x;w)\mu, \quad \mu \in \Delta^{n,k}. \tag{9}$$

Commonly, the model parameters w weight a matrix of features F derived from input x, such that

$$\theta(x;w) = F_\theta(x)w \quad \text{and} \quad \Theta(x;w) = F_\Theta(x)w.$$

For computational reasons, we want to ensure strict convexity of the energy. Towards this end, we first break up the energy into contributions by edges.

Decomposition. From its definition, it follows that the energy can be decomposed as

$$E(\mu|x;w) = \sum_{(i,j)\in\mathcal{E}} E_{ij}(\mu|x;w). \tag{10}$$

Each such pairwise term is of the form

$$E_{ij}(\mu) = \begin{pmatrix}\mu_i\\ \mu_j\end{pmatrix}' \underbrace{\begin{pmatrix}\bar{\theta}_i\\ \bar{\theta}_j\end{pmatrix}}_{\theta_{ij}(x;w)} + \frac{1}{2}\begin{pmatrix}\mu_i\\ \mu_j\end{pmatrix}' \underbrace{\begin{pmatrix}\bar{\Theta}_{ii} & \bar{\Theta}_{ij}\\ \bar{\Theta}'_{ij} & \bar{\Theta}_{jj}\end{pmatrix}}_{\Theta_{ij}(x;w)} \begin{pmatrix}\mu_i\\ \mu_j\end{pmatrix}. \tag{11}$$

If $\bar{\Theta}_{ii}$ and $\bar{\Theta}_{jj}$ are zero, the QP formulation is exact, but the energy is non-convex. To ensure convexity of the global energy, it suffices that $\Theta_{ij}(x;w) \succ 0$.

Convex forms. We now discuss, in increasing order of expressiveness, forms of $\Theta_{ij}(x;w)$ that lead to convex energies and moreover ensure convexity of our learning objective in the model parameters w.

Form 1: A natural approach, followed in the logistic random field (Tappen et al., 2008), is to define

$$\Theta_{ij}(x;w) = F_{ij}(x)'\operatorname{diag}(w_p)F_{ij}(x), \tag{12}$$

where $w_p \in w$ is a component-wise positive weighting vector. By definition, we have $\Theta_{ij}(x;w) \succeq 0$, and if $\operatorname{rank}\{F_{ij}(x)\} \geq 2k$, then $\Theta_{ij}(x;w) \succ 0$ holds. To ensure $w_p > 0$, we can project onto the positive orthant by clipping; but there are only few degrees of freedom.

Form 2: A more powerful approach is to directly learn positive-definite matrices. Assuming $\{W_p\} \in w$ is a set of such matrices, we can use a linear combination

$$\Theta_{ij}(x; w) = \sum_p f_p(x) W_p, \quad f_p(x) > 0. \tag{13}$$

This way, a strictly larger class of interactions can be modeled. To ensure $W_p \succ 0$, we can project it onto the cone of positive-definite matrices at the small cost of a single eigendecomposition.

Form 3: It is often gainful to further exploit the dependency on input x. To this end, we can define

$$\Theta_{ij}(x; w) = F_{ij}(x; \{W_p\}), \tag{14}$$

where F_{ij} is a nonlinear function of the observed input mapping to one of the matrices $W_p \succ 0$. As an example, F_{ij} might evaluate a decision tree on the input, and return the specific matrix W_p stored at the selected leaf. This approach was followed before for Gaussian random fields by Jancsary et al. (2012).

Form 4: Finally, for some applications, fixed-form global interaction matrices Q exist that already model the right semantics, so one can define

$$\Theta(x; w) = w_q Q(x), \quad Q(x) \succ 0, \tag{15}$$

and learn the contribution relative to other terms via a positive scalar $w_q > 0$. Since Q only enters the energy in a matrix-vector product, even dense matrices can be used if they permit efficient multiplication (see §5.4).

4.2. Parameter Estimation

Assume we are given i.i.d. labeled training examples $(\{x_i\}, \{y_i^\star\})$. To facilitate notation, we think of the examples as disconnected components of a single instance, denoted by $(x, y^\star)$ and comprising n nodes. Further, the cost of a misprediction $\hat{\mu}$ is measured by a loss function that decomposes. The loss relative to ground truth $y^\star$ can then be written as $\hat{\mu}'\delta(y^\star)$, e.g.

$$\delta_i(y_i^\star) = [\![\psi_i(y_i^\star) \neq 1]\!] \tag{16}$$

for Hamming loss. Direct minimization of such discontinuous loss functions is infeasible; but following the empirical risk formulation of large margin estimation (Ratliff et al., 2007), we can define the surrogate

$$\xi(y|x;w) = E(\psi(y)|x;w) - \min_{\mu \in \Delta^{n,k}} \left[E(\mu|x;w) - \mu'\delta(y)\right],$$

leading to a convex, regularized estimation problem:

$$\begin{aligned} \underset{w}{\text{minimize}} \quad & \frac{C}{2}\|w\|^2 + \frac{1}{n}\xi(y^\star|x;w) \\ \text{sb.t.} \quad & \Theta_{ij}(x;w) \succ 0 \text{ for all } (i,j) \in \mathcal{E}. \end{aligned} \tag{17}$$

One can verify that $\xi(y|x;w)$ is an upper bound on the loss of the actual prediction obtained from the energy.

Optimization. A convenient property specific to our approach is that for $\Theta_{ij} \succ 0$, the surrogate loss $\xi(y|x;w)$ is differentiable in the model parameters w. To see this, consider the loss augmented inference problem solved in order to compute the surrogate:

$$\hat{\mu}_\delta(x;w) = \underset{\mu \in \Delta^{n,k}}{\operatorname{argmin}} \left[E(\mu|x;w) - \mu'\delta(y)\right]. \tag{18}$$

Since the energy is strictly convex, the minimum is attained uniquely. Then, by Danskin's theorem, the subdifferential of the min function contains a single element (the gradient), obtained by differentiating the inner expression at point $\hat{\mu}_\delta$:

$$\frac{\partial \xi(y|x;w)}{\partial \theta(x;w)} = \psi(y^\star) - \hat{\mu}_\delta$$

and

$$\frac{\partial \xi(y|x;w)}{\partial \Theta(x;w)} = \frac{1}{2}\left[\psi(y^\star)\psi(y^\star)' - \hat{\mu}_\delta\hat{\mu}_\delta'\right],$$

while the gradient with respect to the actual model parameters w follows from the chain rule.

In practice, we found projected quasi-Newton methods (Schmidt et al., 2009) to be effective at solving (17), since cheap closed-form projections ensuring positivity are available for the parameterizations we discussed. Because (17) is convex, we find the global optimum.

4.3. Inference

At test time, given the estimated model parameters $\hat{w}$ and input x, we determine the prediction as

$$\hat{\mu}(x;\hat{w}) = \underset{\mu \in \Delta^{n,k}}{\operatorname{argmin}} E(\mu|x;\hat{w}). \tag{19}$$

Since $\hat{\mu}$ can be fractional, we round to the nearest integral point to find a discrete $\hat{y}$.

For global minimization of (19), we use the spectral projected gradient method (SPG; Birgin et al., 2000), resulting in an efficient iterative scheme that depends only on matrix-vector products $\Theta(x;w)\mu$ as its basic operation. The constraints $\mu \in \Delta^{n,k}$ are handled by independently projecting each μ_i onto a simplex Δ^k, which takes expected linear time (Duchi et al., 2008).

4.4. Relation to Logistic Random Field

The Logistic Random Field (LRF) of Tappen et al. (2008) uses a convex quadratic energy similar to (19), but without constraints $\Delta^{n,k}$. Its parameters w are estimated by minimizing a logistic loss defined directly on the prediction. This can be viewed as a smoothed form of Hamming loss but leads to a non-convex problem, even if the energy is convex in w. Though not used originally, all parameterizations we discussed for our approach are equally applicable to LRF, so we compare to such an extended version in the following.

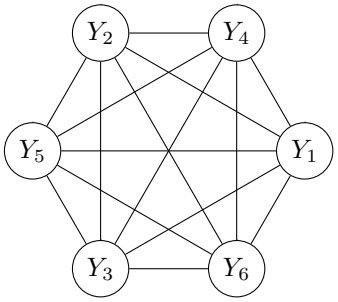


Figure 2. The graphical model for the multi-label classification task, illustrated for the concrete case of the Yeast dataset. Each binary variable Y_i encodes whether label i shall be one of the labels assigned to the example.

Table 1. Average test loss in multi-label classification, along with standard error of the mean. The results for M3N and LP-M3N are reported by Finley & Joachims (2008).

Data	M3N	LP-M3N	QP-M3N	LRF
Scene	10.06±.26	10.49±.27	10.48±.36	10.98±.33
Yeast	20.23±.53	20.49±.54	20.19±.45	29.05±.48

5. Applications and Experiments

We now assess the benefits of our approach on four structured prediction tasks that lead to intractable inference and training, comparing *QP-M3N:* large margin learning with convex quadratic programming relaxations (our approach); *LP-M3N:* large margin learning with linear programming relaxations; *LRF:* logistic random field of Tappen et al. (2008), with quadratic regularization of the model parameters; and finally *M3N:* exact max-margin Markov networks, which are only feasible for small instances.

All experiments are performed on recent Intel Xeon machines with 16GB of main memory. Depending on the method and application, different iterative solvers are used for inference and parameter estimation, but we always tried to ensure that the convergence criteria lead to a fair comparison in terms of both predictive accuracy and computational efficiency.

5.1. Experiment 1: Multi-label Classification

The problem of multi-label classification is to assign a *subset* of labels $\mathcal{Y} \subseteq \{1,\ldots,k\}$ to each example. This task can be expressed equivalently via k binary variables $y_i \in \{0,1\}$ that each encode whether or not label i shall be assigned to an example. These binary variables are correlated since some label combinations are more likely than others. We follow Finley & Joachims (2008) and encode the dependencies via a graphical model with dense pairwise connectivity (see Figure 2).

`He`	`remains`	`chief`	`executive`	`officer`	`.`
PRP	VBZ	JJ	JJ	NN	.
B-NP	B-VP	B-NP	I-NP	I-NP	O

Figure 3. An exemplary sentence of the joint PoS-tagging and chunking task, taken from CoNLL-2000 data. The PoS chain uses 44 labels, while 23 labels are used for chunking.

We evaluate on the Scene and Yeast datasets, which were used by Finley & Joachims (2008) without any pre-processing, allowing us to replicate the experiment exactly. Besides the positivity constraints on the pairwise matrices in the QP-M3N and LRF systems, as in (13), our setup is equivalent to theirs: Single-node potentials are obtained as dot products of model parameters and input features, while the pairwise terms learn a co-occurrence bias for label combinations. For each dataset and learning method, we choose from 14 different settings of regularization parameter C via 10-fold cross validation on the training data. The best value of C is then used to train on the whole training data, and we report the average normalized Hamming loss on test examples (this is the loss all methods more or less directly *try* to minimize during training).

Table 1 shows the new results obtained for QP-M3N and LRF, alongside the results of Finley & Joachims (2008). The standard error of the mean is indicated next to the loss. Since the graphical models are tiny, it is possible to compare against an exactly trained M3N. As one can see, QP-M3N compares favorably on both datasets, achieving lower test loss than LP-M3N and even M3N on the Yeast dataset. LRF, on the other hand, fails on the Yeast dataset. It is unclear to us exactly why this is the case; we tried to tweak various implementation details of the method in order to achieve better results, to no avail. Presumably, non-convexity of the training objective used by LRF is to blame, although this was not in general a problem.

5.2. Experiment 2: PoS-Tagging and Chunking

The multi-label classification problem uses very small graphical models, allowing for rigorous evaluation of the predictive performance, but preventing realistic comparisons in terms of computational efficiency. We now consider a different problem, joint Part-of-Speech tagging and phrase chunking, which already leads to inference problems of more realistic size and difficulty. The goal is to identify the lingustic category of each word in a sentence, often referred to as a "part of speech". Moreover, phrase boundaries and types shall be identified. This can be formulated as a tagging problem on two chains, as illustrated in Fig. 3. Again, the correlations between labels can be exploited.

Table 2. Test loss on the joint PoS-tagging and chunking task, using a *strong* unary classifier achieving a loss of 3.78.

C	M3N	LP-M3N	QP-M3N	LRF
10^{-0}	13.88	15.18	13.54	3.87
10^{-3}	3.58	3.54	3.76	3.87
10^{-6}	$\sim$	3.48	3.56	3.81
10^{-9}	$\sim$	3.41	3.56	4.03

Table 3. Test loss on the joint PoS-tagging and chunking task, using a *weak* unary classifier achieving a loss of 13.17.

C	M3N	LP-M3N	QP-M3N	LRF
10^{-0}	22.15	28.60	21.06	5.53
10^{-3}	6.22	6.14	7.98	6.61
10^{-6}	$\sim$	5.74	6.64	6.02
10^{-9}	$\sim$	5.74	6.61	5.95

Our experiments are performed on the CoNLL-2000 shared task (Tjong Kim Sang & Buchholz, 2000), comprising 8,936 training sentences and 2,012 test sentences. As a baseline, we train two local SVM classifiers predicting each label independently. We use sparse binary features similar to Sutton et al. (2004) that check for the occurrence of words in a window around the current token. The first classifier, called *strong*, uses a window size of ± 3 tokens, while the second one, *weak*, is restricted to a window of ± 1 tokens. These classifiers are trained on the training data using crossvalidation and achieve an average normalized Hamming loss of 3.78 and 13.17 on the test data.

In order to take into account correlations between the output variables, we specify a graphical model similar to Sutton et al. (2004), with a pairwise factor defined between adjacent words *within* each label chain, as well as a pairwise factor defined *between* the two chains for each word, resulting in a grid structure. Pairwise factors are specified as in (13), with factors of the same spatial type sharing the same model parameters. The unary factors are derived from the confidence scores predicted by either the *strong* or the *weak* local SVM classifier, creating two different scenarios.

Exact inference in this model is still feasible by forming junction trees, which typically results in cliques involving no more than three variables. For moderately small values of regularization parameter C, this allows us to train exact M3N models using bundle methods (Teo et al., 2010). For LP-M3N, we follow Hazan & Urtasun (2010) and slightly smooth the LP using a concave entropy approximation ($\varepsilon = 10^{-2}$), resulting in a differentiable training objective that can be solved efficiently for any value of C. Inference is done using convex belief propagation (Hazan & Shashua, 2010). We use our own implementation of these methods.

Let us first discuss the results obtained by using the *weak* local classifier. The test loss of models trained on the training data using a range of regularization parameters C is shown in Table 2. Here, LP-M3N performs comparably to an exact M3N, while QP-M3N is slightly worse but still improves significantly over the baseline classifier. LRF, on the other hand, achieves worse results than the baseline classifier. Moreover, regularization is less predictable than for the other models. The standard error of the test loss ranges from 0.1 to 0.2, so these results are rather conclusive.

Table 4. Computational efficiency in joint PoS-tagging and chunking, for the best value of C and *weak* unaries.

	M3N	LP-M3N	QP-M3N	LRF
Train	58h	11h	12h	38h
Test	12 $^{\text{sent}}/_{\text{s}}$	11 $^{\text{sent}}/_{\text{s}}$	313 $^{\text{sent}}/_{\text{s}}$	165 $^{\text{sent}}/_{\text{s}}$

Using *weak* baseline unaries, bigger improvements can be expected from structured models. The results in Table 3 mostly follow the previous discussion, with the surprising exception that LRF achieves the strongest results. This suggests that the restriction to convex quadratic energies does not *per se* limit the expressiveness of a model on this task. The slightly worse results achieved by QP-M3N must then stem either from interactions with large margin estimation *or* the additional simplicial constraints in the energy.

Table 4 compares computation time. Inference in QP-M3N is very efficient, yielding predictions more than an order of magnitude faster than LP-M3N and even twice as fast as LRF, for which we use conjugate gradient. Exact inference in M3N is as fast as solving the LP relaxation using convex message passing, since the prediction can be readily obtained from a junction tree. Training time is somewhat less informative; the inference subproblem does not strongly dominate the cost here, so it mostly depends on the number of iterations required to optimize the model parameters.

5.3. Experiment 3: Inpainting of Characters

We now consider a task requiring conditional pairwise interactions that strongly depend on the observed input. The goal is to inpaint the occluded parts of Chinese characters (cf. Figure 4, occlusions in gray). This problem was first considered by Nowozin et al. (2011); the data consists of 300 training and 100 test images. Following the original experiment, we visualize predictions on a version of the data with larger occlusions and report predictive accuracy on smaller occlusions.

Figure 4. Inpainting of Chinese characters. Typical predictions by various system configurations are shown.

The graphical model we use is identical to Nowozin et al. (2011). In particular, we instantiate pairwise factors according to 32 types of spatial offsets relative to each pixel, such that each variable is connected to 64 other variables in its neighborhood. The potentials of a pairwise factor are determined by its type: Each type has an associated decision tree that performs feature checks on the input image relative to the coordinates of the factor. Leaf nodes store the positive-definite interaction matrices, and so the path taken through the tree determines the effective interaction for each factor, as in (14). We also use one unary type.

Learning follows a two-step iterative scheme that alternates between introducing new model parameters by splitting tree nodes, and optimization of the current model parameters according to the learning objective. Tree nodes are split with the goal of achieving the largest possible descent in the objective function. This approach was introduced by Jancsary et al. (2012) in order to train *Regression Tree Fields* (RTFs), but it is equally applicable to QP-M3N and LRF. We train unary trees to a depth of 10 and pairwise trees to a depth of 4. Deeper trees lead to overfitting.

Besides Regression Tree Fields (RTFs), we compare against a random forest (RF), as well as *Decision Tree Fields* (DTF), introduced by Nowozin et al. (2011). RTFs are based on a Gaussian conditional random field while DTFs use a discrete random field. Both methods are trained by maximizing the pseudolikelihood (Besag, 1975) of the data. Predictions are obtained from RTF using conjugate gradient (CG), similarly to LRF. In contrast, DTF must solve an intractable min-sum problem. One option is to solve the LP relaxation instead; but, as shown in Table 6, this is still expensive. TRW-S (Kolmogorov, 2006), specialized for binary problems, is over an order of magnitude slower than inference in QP-M3N and the approaches based on a Gaussian random field. While TRW-S is recognized as one of the most efficient solvers for general binary labeling problems, it must update $O(2k|\mathcal{E}|)$ variables, versus $O(k|\mathcal{V}|)$ variables in approaches based on a quadratic energy. This makes a big difference in these rather densely connected graphs. Even worse, while TRW-S aims to solve the LP relaxation, it often gets stuck. Better results for DTF are obtained by using simulated annealing, which is even slower. End-to-end discriminative training using the LP relaxation, as in LP-M3N, is conceivable, but would be extremely expensive, since inference clearly dominates the computational cost of training in this task. Training of QP-M3N and LRF took 20 and 28 hours, respectively.

Table 5. Test loss on the Chinese characters inpainting task. The numbers for RF, DTF and RTF are reported by Nowozin et al. (2011) and Jancsary et al. (2012).

C	QP-M3N	LRF	RF	RTF	DTF
10^{-0}	43.22	20.95	32.26	22.45	23.99
10^{-3}	22.36	22.35			
10^{-6}	24.01	21.48			
10^{-9}	23.57	20.64			

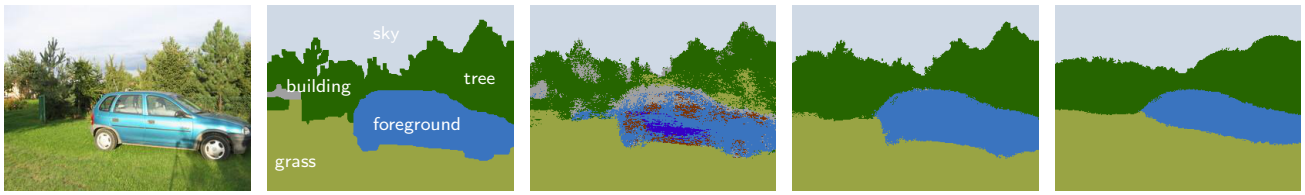
Table 6. Chinese characters: Computational efficiency at test time vs. inference algorithm used (seconds per image).

Algorithm	QP-M3N	LRF	RTF	DTF
SPG	0.25s			
CG		0.22s	0.24s	
TRW-S/LP				4.79s
Annealing				≈20s

Table 5 lists the predictive accuracy of the competing approaches, again measured in terms of average normalized Hamming loss. Both QP-M3N and LRF are competitive with the state of the art; LRF in particular improves considerably on previously published results. Again, this is rather surprising. In any case, Hamming loss is perhaps not the most appropriate performance measure on this task. Of equal interest is whether the predictions look like plausible Chinese characters. Typical predictions are shown in Figure 4.

5.4. Experiment 4: Semantic Segmentation

Finally, we show how fixed-form interaction matrices can be employed in our model, as in (15). The goal in semantic segmentation is to correctly identify the parts and foreground objects of a scene, which can be formalized as assigning one of k labels to each pixel of an image (Fig. 5). It is desirable for segmentations to expose a certain level of connectedness. To this end, one can use an affinity matrix based on the pixels' similarity in appearance. We use the *matting Laplacian*,



(a) Input (b) Ground truth (c) Random forest (d) QP-M3N (e) LRF

Figure 5. Semantic segmentation of a previously unseen image. QP-M3N and LRF use a matting Laplacian with $r = 16$.

Table 7. Segmentation results for varying window radii r of the matting Laplacian. A random forest baseline achieves a test loss of 30.62. Images are of size 320×240, 8 labels.

	Test loss		*Inference time*	
r	QP-M3N	LRF	QP-M3N	LRF
2	28.90	29.11	1.3 s/image	2.1 s/image
4	28.70	29.23	1.2 s/image	1.8 s/image
16	28.42	30.69	0.9 s/image	2.3 s/image

defined by Levin et al. (2008) as a sum of matrices

$$L(x) = \sum_{i \in \mathcal{V}} A_i(x), \tag{20}$$

each of which stores the affinities among pixels inside a window of radius r. Since L is positive semi-definite, we can incorporate it into the quadratic energies of QP-M3N and LRF. Importantly, L allows for efficient multiplication at constant cost in its window radius r (He et al., 2010). Since inference in QP-M3N and LRF only requires such products, full connectivity in arbitrarily large windows can be modeled. We do not know how to achieve this efficiently in the LP relaxation.

In our experiment, we use the matting Laplacian to refine segmentations obtained from a random forest. We perform 5-fold crossvalidation on the 715 images in the scene understanding dataset of Gould et al. (2009). For each fold, a random forest is first trained on the training portion of the data. The QP-M3N and LRF models then incorporate the predictions of the random forest as weighted unary features and learn their relative importance versus a matting Laplacian term. Due to the small number of model parameters, the structured models are trained only on one third of the training portion of each fold, and we do not regularize. All models are evaluated on the test portion of each fold, and the results are then averaged over the folds.

Table 7 shows that the gains of QP-M3N over the random forest grow with increasing window radius r, while the computational cost slightly decreases. In contrast, LRF even fails to improve over the baseline at $r = 16$. Empirically, we find that LRF over-smoothes its predictions (see Figure 5). It is also less efficient: Training takes 10 hours versus 2 hours for QP-M3N.

6. Discussion and Summary

Based on our findings, is it possible to recommend one method over the others? Perhaps most closely related to learning using convex QP relaxations (QP-M3N), which we proposed in this paper, is the Logistic Random Field (LRF) of Tappen et al. (2008). It achieved better results than QP-M3N in two cases; but it also failed completely in three scenarios, not even improving on a trivial baseline. We believe that this is due to its non-convex learning objective, and also the fact that regularization of the model is more difficult. In contrast, QP-M3N always improved reliably on the baseline, and it is at least as efficient. For this reason, we would advise against the use of LRF in general.

The closest competitor of QP-M3N is large margin learning with LP relaxations (LP-M3N). It achieved better results than QP-M3N in one of the two tasks where a direct comparison was possible. On the other hand, in our experiments, it was at least an order of magnitude faster to solve the convex QP relaxation. This is not surprising: The LP relaxation involves significantly more variables and a more difficult constraint set. In practice, the computational efficiency of QP-M3N enables end-to-end discriminative training where LP-M3N is simply too expensive. QP-M3N is also more flexible, in that it allows to incorporate dense interaction matrices efficiently.

A drawback of our approach is that the limitations in the expressiveness of convex QP relaxations are not yet as well understood as those of other classes of energies that can be minimized efficiently, such as submodular ones. Nonetheless, we hope to have demonstrated that convex QP relaxations are widely applicable in practice and computationally efficient. Furthermore, they can be trained in a principled manner in a large margin framework and yield competitive predictive accuracy. For these reasons, we believe that convex QP relaxations deserve further attention.

Acknowledgments

We would like to thank Toby Sharp for sharing efficient code for multiplying by the matting Laplacian.

References

Besag, J. Statistical analysis of non-lattice data. *The Statistician*, 24(3), 1975.

Birgin, E., Martínez, J., and Raydan, M. Nonmonotone spectral projected gradient methods on convex sets. *SIAM Journal on Optimization*, 10, 2000.

Domke, J. Parameter learning with truncated message-passing. In *CVPR*, 2011.

Duchi, J., Shalev-Shwartz, S., Singer, Y., and Chandra, T. Efficient projections onto the ℓ_1-ball for learning in high dimensions. In *ICML*, 2008.

Finley, T. and Joachims, T. Training structural SVMs when exact inference is intractable. In *ICML*, 2008.

Franc, V. and Savchynskyy, B. Discriminative learning of max-sum classifiers. *JMLR*, 9, 2008.

Globerson, A. and Jaakkola, T. Fixing max-product: Convergent message passing algorithms for MAP LP-relaxations. In *NIPS*, 2007.

Gould, S., Fulton, R., and Koller, D. Decomposing a scene into geometric and semantically consistent regions. In *ICCV*, 2009.

Hazan, T. and Shashua, A. Norm-product belief propagation: Primal-dual message-passing for approximate-inference. *IEEE Trans. Inf. Theory*, 56, 2010.

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

He, K., Sun, J., and Tang, X. Fast matting using large kernel matting Laplacian matrices. In *CVPR*, 2010.

Jancsary, J., Nowozin, S., Sharp, T., and Rother, C. Regression tree fields – an efficient, non-parametric approach to image labeling problems. In *CVPR*, 2012.

Kolmogorov, V. Convergent tree-reweighted message passing for energy minimization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28, 2006.

Komodakis, N. Efficient training for pairwise higher order CRFs via dual decomposition. In *CVPR*, 2011.

Koval, V. and Schlesinger, M. Two-dimensional programming in image analysis problems. *USSR Academy of Science, Automatics and Telemechanics*, 8, 1976.

Kulesza, A. and Pereira, F. Structured learning with approximate inference. In *NIPS*, 2007.

Kumar, M., Kolmogorov, V., and Torr, P. An analysis of convex relaxations for MAP estimation of discrete MRFs. *JMLR*, 10, 2009.

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

Levin, A., Rav-Acha, A., and Lischinski, D. Spectral matting. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30, 2008.

Martins, A., Smith, N., and Xing, E. Polyhedral outer approximations with application to natural language parsing. In *ICML*, 2009.

Meshi, O., Sontag, D., Jaakkola, T., and Globerson, A. Learning efficiently with approximate inference via dual losses. In *ICML*, 2010.

Nowozin, S., Rother, C., Bagon, S., Sharp, T., Zao, B., and Kohli, P. Decision tree fields. In *ICCV*, 2011.

Ratliff, N., Bagnell, A., and Zinkevich, M. (Online) subgradient methods for structured prediction. In *AISTATS*, 2007.

Ravikumar, P. and Lafferty, J. Quadratic programming relaxations for metric labeling and Markov random field MAP estimation. In *ICML*, 2006.

Ravikumar, P., Agarwal, A., and Wainwright, M. Message-passing for graph-structured linear programs: Proximal methods and rounding schemes. *JMLR*, 11, 2010.

Savchynskyy, B., Schmidt, S., Kappes, J., and Schnörr, C. Efficient MRF energy minimization via adaptive diminishing smoothing. In *UAI*, 2012.

Schmidt, M., van den Berg, E., Friedlander, M., and Murphy, K. Optimizing costly functions with simple constraints: A limited-memory projected quasi-Newton algorithm. In *AISTATS*, 2009.

Stoyanov, V., Ropson, A., and Eisner, J. Empirical risk minimization of graphical model parameters given approximate inference, decoding, and model structure. In *AISTATS*, 2011.

Sutton, C. and McCallum, A. Piecewise training for structured prediction. *Machine Learning*, 77, 2009.

Sutton, C., Rohanimanesh, K., and McCallum, A. Dynamic conditional random fields: Factorized probabilistic models for labeling and segmenting sequence data. In *ICML*, 2004.

Tappen, M., Samuel, K., Dean, C., and Lyle, D. The logistic random field – a convenient graphical model for learning parameters for MRF-based labeling. In *CVPR*, 2008.

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

Taskar, B., Lacoste-Julien, S., and Jordan, M. Structured prediction, dual extragradient and bregman projections. *JMLR*, 7, 2006.

Teo, C., Vishwanathan, S.V.N., Smola, A., and Le, Q. Bundle methods for regularized risk minimization. *JMLR*, 11, 2010.

Tjong Kim Sang, E. and Buchholz, S. Introduction to the CoNLL-2000 shared task: Chunking. In *CoNLL*, 2000.

Wainwright, M. Estimating the "wrong" graphical model: Benefits in the computation-limited setting. *JMLR*, 7, 2006.

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