This paper was submitted as a final project report for CS6424/ECE6424 *Probabilistic Graphical Models and Structured Prediction* in the spring semester of 2016.

The work presented here is done by students on short time constraints, so it may be preliminary or have inconclusive results. I encouraged the students to choose projects that complemented or integrated with their own research, so it is possible the project has continued since this report was submitted. If you are interested in the topic, I encourage you to contact the authors to see the latest developments on these ideas.

Bert Huang
Department of Computer Science
Virginia Tech

Efficient Training of MRFs with Latent Variable using Paired-Dual Learning

Elaheh Raisi
Department of Computer Science, Virginia Tech
elaheh@vt.edu

Bert Huang
Department of Computer Science, Virginia Tech
bhuang@vt.edu

Abstract

In many real-world applications of learning, the available data are incomplete, which poses significant computational challenges. In many proposed methods to learn the model containing latent variable, we need repeated inference to iteratively update parameters. Inferences are usually expensive for large models. In this study, we propose a framework that quickly trains Markov random fields with latent variables by avoiding repeated inferences. We used a variational learning objective that substitutes belief propagation dual problems for two corresponding inference problems, augmented with Bethe entropy. We demonstrate the effectiveness of the proposed method in the task of image segmentation, showing that regarding training time, our approach is superior to traditional methods, converging faster to the optimal solution.

1 Introduction

In many applications such as natural language processing, computational biology, and computer vision, labeling all the data is a very expensive process and often impractical. For example, in computer vision, gathering annotated data is very difficult; or in some scientific applications, obtaining the labels involves repeated experiments that may be hazardous; in drug prediction, deriving active molecules of a new drug involves expensive expertise that may not even be available [11]. In the presence of weakly labeled data, some parts of data distribution are hidden; by using latent variables we can capture the hidden structure of the distribution. Including latent variables in probabilistic graphical model, however, is challenging due to the computational costs imposed by the unknown values of latent variables.

Several approaches have been developed to deal with learning latent variables. The traditional method is expectation maximization [10] in which we maximize the posterior probability of the parameters given the data, marginalizing over latent variables. EM iteratively alternates between an E step, computing an expectation of the likelihood by including the latent variables as if they were observed, and an M step, maximizing the expected likelihood found on the E step. The resulting parameters in the M step are used to begin another E step, and the process is repeated. Since computing the likelihood of the observed labels, marginalizing over the latent variable is expensive, variational methods of EM were introduced [7]. In variational EM, we iteratively minimize the KL divergence to the empirical distribution, and estimate the expectation of the latent variables.

EM and variational EM methods are expensive because they need repeated inference to update the parameters. Inference by itself is not an easy process. For large models, inferences such as belief propagation and Gibbs sampling need many iterations to converge. So, overcoming the need for

Probabilistic Graphical Models and Structured Prediction Final Project Report. Spring 2016, Virginia Tech.

repeated inference for such complex models are necessary.
Taskar et al. and Meshi et al. in [21, 13] speed up learning by applying the dual of inference to make a joint convex minimization for fully supervised data. Schwing et al. [20] used the same idea in the case when we have latent variables. Our proposed method is based on the methodology introduced by Bach et al. [3]. They aimed to address the computational bottleneck incurred by continuous latent variable models for hinge-loss Markov random fields (HL-MRFs) [2]. Their proposed framework, paired-dual learning, quickly trains HL-MRFs with latent variables by avoiding repeated inferences. Paired-dual learning uses an equivalent variational learning objective that substitutes dual problems (ADMM [5]) for the two corresponding inference problems, augmented with entropy surrogates to make the learning problem well formed. By computing the gradient of paired-dual learning objective with respect to the parameters using the intermediate states of inference, they formulate a fast, block-coordinate joint optimization.
In this study, instead of HL-MRF, we apply the proposed method for discrete Markov random field network. Besides, we use belief propagation for dualization of inference problems, augmented with Bethe approximation. Bethe entropy approximation is a substitution for negative conjugate dual [23]. They are computed by marginal distributions on the node and edges, which correspond to the mean parameters.
For evaluation, we apply our method to image segmentation task. In image segmentation, each pixel is classified into a semantic category. Since annotating data is a very expensive process, we weakly annotated data. So, our task is to train a model that can finally label the unknown pixels. In our experiments, we show that the time for learning MRF with latent variable is reduced significantly compared to EM and subgradient. Our proposed method, we call it Bethe PDL, converges much faster than two baselines, while having the same accuracy as slow traditional methods.

2 Related Work

Various approaches have been proposed to learn the models with latent variables. [10, 12] used marginal MAP inference by averaging over the hidden variables, and then optimizing over the variables of direct interest. In many domains, marginal MAP can provide significant improvement over joint MAP estimation, which jointly optimizes hidden and output variables. Nevertheless, marginal MAP can be NP-hard even when the underlying graphical model is a tree-structured [10]. Liu et al. [12] proposed an efficient variational algorithm that approximately solves marginal MAP.
Hidden-state conditional random field (HCRF) [18] is an extension of CRF to include hidden variables. It learns a set of latent variables conditioned on local features. Observations need not be independent and may overlap in space and time. HCRF model combines the ability of CRFs to use dependent input features and the ability of HMMs to learn latent structure. They have many applications including but not limited to object recognition [17] and gesture recognition [24]
The hidden-unit conditional random fields [22] is a generalization of conditional random fields (CRFs) in which binary stochastic hidden units appear between the data and the labels. These units are conditionally independent given the data and the label sequence. Unlike CRFs, they can represent nonlinear dependencies at each frame. Poon et al. [16] introduced a deep architecture, sum-product networks, which are directed acyclic graphs with variables as leaves, sums and products as internal nodes, and weighted edges. Interior nodes in an SPN can be interpreted as latent variables. They state that all tractable graphical models can be cast as SPNs, and then learning algorithms for SPNs, based on backpropagation and EM was proposed.
The other notable class of learning with latent variables is latent structured support vector machine (LSSVM) [28]. LSSVM is an extension of the Structural SVM framework to include latent variables. They identify a formulation for which there exists an efficient algorithm to find a local optimum using the Concave-Convex procedure. The Concave-Convex Procedure [29] is a general framework for minimizing non-convex functions, which falls into the class of Difference of Convex programming. LSSVM has applications in a wide range of areas such as object detection [30], human action recognition [25], and link prediction [26]
Marginal structured SVM [15] was also proposed for structured prediction with hidden variables, which inherits the general advantages of structured SVM. In this model, the uncertainty of the hidden variables is considered by incorporating marginal MAP inference that averages over the possible hidden states. They also introduced a unified framework that includes both their method and LSSVM and HCRFs methods as special cases. The main drawback of these methods is that they require repeated inference which a very expensive process. To lift this restriction, for fully-supervised

learning, large-margin methods can use the dual of loss-augmented inference to form a joint con-vex minimization [21, 13]. Schwing et al. [20] extended this idea to latent variable learning for discrete MRF. They proposed a unified framework for structured prediction with latent variables, which includes hidden conditional random fields and latent structured support vector machines as special cases. They describe a local entropy approximation for their formulation by dualizing one of the two inference subroutines, and derive an efficient message-passing algorithm that is guaranteed to converge. Consequently, they speed up the learning of discrete models with latent variables.
Bach et al. [3] introduced a framework called paired-dual learning. In order to make the training process faster, they used a tractable entropy surrogate and avoid repeated inferences. They formulate an objective with a pair of dual inference problems using ADMM [5]. To compute gradients of the learning objective, instead of full inference, they used incomplete dual inference optimizations. They showed that paired-dual learning is able to train accurate models in a small fraction of the time required by traditional algorithms. On their study, they focus on hinge-loss Markov random fields (HL-MRFs) [2], a class of probabilistic graphical models, which represent structured domains with continuous variables.
There are also some other studies to deal with latent variable in HL-MRF. HL-MRF with latent variable are trained for task such as group detection in social media [4] online-education analytics [19] and automobile-traffic modeling [6]. Using dual inferences in learning objective function has been taken into consideration in many works. For fully-supervised settings, Taskar et al. [21] dualize inference problem as part of large-margin learning, making a joint quadratic program. [13] use dual decomposition for LP relaxations of inference in discrete graphical models. [20] extend this idea to latent-variable models by dualizing one of the two inference problems and passing messages corresponding to the discrete states. Domke et al. [9] used dualization as part of a technique to reduce structured prediction for non-structured logistic regression.
Our work is similar to the [3] since we are formulating a learning objective function using a pair of dual inferences. However, we apply belief propagation dualization. In addition, our study concentrates on discrete random Markov field, which is a general graphical model used in many applications.

3 Background

3.1 Belief Propagation on pairwise MRF

Belief propagation [14] is a message passing algorithm proposed by Pearlin, for performing inference on graphical models. It calculates the marginal distribution for each unobserved node, conditioned on the observed nodes. It is proved that belief propagation exactly computes marginals on tree graphs. However, belief propagation has empirically demonstrated to be effective on loopy graphs. The belief propagation algorithm works by sending messages along the edges of the graph. The beliefs are equal to the marginal probabilities for graphs. BP is an iterative process in which neighboring variables talk to each other, passing messages regarding different states a variable can take. After some amount of iterations, the conversation will converge so that the marginal probabilities or "beliefs" of all the variables can be determined. By considering pairwise MRF, the probability distribution will be:

$$p(X) = \frac{1}{Z} \prod_{i=1}^{N} \phi_i \prod_{<i,j>} \psi_{i,j}(x_i, x_j) \tag{1}$$

where ϕ_i are unary factors and $\psi_{i,j}$ are pairwise factors. The second product is done over every neighboring pair of nodes in the graph. $m_{ij}(x_j)$ indicates the message from node i to node j. It shows that node i belief about node j to have the state x_j. If this value is high, it means that node i believes the marginal value of node j should be high. The messages are updated using:

$$m_{ij}(x_j) = \sum_{x_i} \psi_{i,j}(x_i, x_j)\phi_i(x_i) \prod_{k \in N(i) \setminus j} m_{ji}(x_i) \tag{2}$$

There are two methods for updating the messages. One is synchronous, which updates all the messages in parallel; and the other one is asynchronous, which updates one message at a time. When the messages have converged, we can compute the beliefs for each node:

$$b_i(x_i) \propto \phi_i(x_i) \prod_{k \in N(i)} m_{ki}(x_i) \tag{3}$$

By normalizing the belief, we can approximate the marginal probabilities. We described the sum-product form of BP algorithm. There is another procedure which uses max-product to estimate the state configuration with maximum probability.

3.2 Variational Learning with Latent Variables

When there are latent variables, we usually maximize the marginal likelihood of the labels given observed variable, marginalizing out over latent variables. If we consider $\mathbf{x}$ as observation, $\mathbf{y}$ as labels which are available during training, $\mathbf{z}$ as latent variable, and θ as model parameters, the marginal likelihood can be written as:

$$p(\mathbf{y}|\mathbf{x};\theta) = \sum_{\mathbf{z}} p(\mathbf{y},\mathbf{z}|\mathbf{x},\theta) \tag{4}$$

The probability distribution $p(\mathbf{u};\theta)$ with the exponential family form with exponential parameter θ and sufficient statistics ϕ is:

$$p(\mathbf{u};\theta) = \frac{1}{Z(\theta)} exp(\theta^T \phi(\mathbf{u})) \tag{5}$$

where $Z(\theta)$ is partition function or normalized function:

$$Z(\theta) = \sum_{\mathbf{u}} exp(\theta^T \phi(\mathbf{u})) \tag{6}$$

Using conditional probability $p(\mathbf{y}|\mathbf{x};\theta)$ can be written as:

$$p(\mathbf{y}|\mathbf{x};\theta) = \frac{p(\mathbf{y},\mathbf{z}|\mathbf{x};\theta)}{p(\mathbf{z}|\mathbf{x},\mathbf{y};\theta)} \tag{7}$$

Therefore, plugging the aforementioned definitions to $p(\mathbf{y}|\mathbf{x};\theta)$, the logarithm of probability distribution $p(\mathbf{y}|\mathbf{x};\theta)$ will be:

$$\log p(\mathbf{y}|\mathbf{x};\theta) = \log p(\mathbf{y},\mathbf{z}|\mathbf{x};\theta) - \log p(\mathbf{z}|\mathbf{x},\mathbf{y};\theta) = \log Z(\mathbf{x},\mathbf{y};\theta) - \log Z(\mathbf{x};\theta) \tag{8}$$

Using variational methods for log partition function Z [10]:

$$\log Z = \max_{\rho \in \Delta(\mathbf{u})} E_\rho[\theta^T \phi(\mathbf{u})] + H(\rho) \tag{9}$$

In equation 9, the first term is energy function and the second term is entropy. Plugging equation 9 to equation 8 will result in:

$$\min_{\rho \in \Delta(\hat{\mathbf{y}},\hat{\mathbf{z}})} \max_{q \in \Delta(\mathbf{z})} E_q[\theta^T \phi(\mathbf{x},\mathbf{y},\mathbf{z})] + H(q) - E_\rho[\theta^T \phi(\mathbf{x},\hat{\mathbf{y}},\hat{\mathbf{z}})] - H(\rho) \tag{10}$$

ρ is a joint distribution over the $\hat{\mathbf{y}}$ and $\hat{\mathbf{z}}$ variables from the space of all joint distributions $\Delta(\hat{\mathbf{y}},\hat{\mathbf{z}})$, q is a conditional distribution over the $\mathbf{z}$ variables from the space of all conditional distributions $\Delta(z)$, and H is the entropy. The final regularized optimization likelihood looks like:

$$\max_{\theta} \min_{\rho \in \Delta(\hat{\mathbf{y}},\hat{\mathbf{z}})} \max_{q \in \Delta(\mathbf{z})} E_q[\theta^T \phi(\mathbf{x},\mathbf{y},\mathbf{z})] + H(q) - E_\rho[\theta^T \phi(\mathbf{x},\hat{\mathbf{y}},\hat{\mathbf{z}})] - H(\rho) + \frac{\lambda}{2}||\theta||^2 \tag{11}$$

where λ is a regularization parameter. To solve optimization equation 11, we can use expectation maximization. To do so, first we solve the conditional inference over $\mathbf{z}$. This process looks like an expectation step. Then by fixing $\mathbf{z}$, we solve the outer max-min over $\mathbf{y}$, $\mathbf{z}$, θ; this process resembles the maximization step. These two steps are repeated until convergence. The other traditional approach to solve this optimization problem is subgradient in which we compute subgradients of the outer maximization over θ by solving the inner min-max and differentiating. For these two approaches, we require at least two inferences per iteration, which is a very expensive process especially for large models.

4 Paired-Dual Learning on MRF

Optimizing the variational learning objective of equation 11 is intractable. To make this learning objective tractable, we replace the inference with dual inference. Using belief propagation dualization,

the primal objective will be replaced by its duals. We know that log partition function can be written as [23]:

$$\log Z(\theta) = sup_{\tau \in T}\{< \theta, \tau > -A^*(\tau)\} = sup_{\tau \in T}\{< \theta, \tau > +H_{Bethe}(\tau)\} \tag{12}$$

Where τ are pseudomarginals which can be interpreted as beliefs. The Bethe entropy is an approximation of the exact dual function $A^*(\tau)$ which will be calculated using:

$$H(\tau) = -\sum_{s \in V}(d_s - 1)H_s(\tau_s) + \sum_{(s,t) \in E} H_{st}(\tau_{st}) \tag{13}$$

where d_s indicates the number of neighbors of node s. τ_s and τ_{st} are pseudomarginals corresponding to node s and edge (s,t) respectively. $H_s(\tau_s)$ is singleton entropy and $H_{st}(\tau_{st})$ is joint entropy defined in [27]. Using sum-product message passing algorithm (belief propagation) we can solve optimization problem 12.
To make the connection between variational problem 12 and the sum-product algorithm, we associate some Lagrange multiplier with constraints related to sum-product method. We consider λ_{ss} to be a Lagrange multiplier associated with the normalization constraint $C_{ss}(\tau) = 0$:

$$C_{ss}(\tau) = 1 - \sum_{x_s} \tau_s(x_s) \tag{14}$$

and $\lambda_{ts}(x_s)$ be a Lagrange multiplier associate with the constraint $C_{ts}(x_s; \tau) = 0$:

$$C_{ts}(x_s; \tau) = \tau_s(x_s) - \sum_{x_t} \tau_{st}(x_s, x_t) \tag{15}$$

These Lagrange multipliers are closely related to sum-product messages: $M_{ts}(x_s) \propto exp(\lambda_{ts}(x_s))$. So, we can consider the Lagrangian accord with the Bethe variational problem 12:

$$\log Z(\theta) =< \theta, \tau > +H_{Bethe}(\tau) + \sum_{s \in V} \lambda_{ss} C_{ss}(\tau) \tag{16}$$

$$+ \sum_{(s,t) \in E} \left[\sum_{x_s} \lambda_{ts}(x_s) C_{ts}(x_s; \tau) + \sum_{x_t} \lambda_{st}(x_t) C_{st}()x_t; \tau \right] \tag{17}$$

The final optimization problem using dualization will be:

$$\max_{\theta} \; min_{\rho \in \Delta(\hat{y}, \hat{z})} \max_{q \in \Delta(z)} < \theta, \tau_q > +H(\tau_q) - < \theta, \tau_\rho > -H(\tau_\rho) + \frac{\lambda}{2}||\theta||^2 + Constraints \tag{18}$$

The last term "Constraints" are the constraints we defined which are included in 16. Our final optimization problem is equation 18. As mentioned earlier, a naive approach to solve this optimization problem is to use subgradient of outer maximization and then solving the inner joint optimization. Another approach can be first solving inference for z, then solving the outer joint optimization and repeat this process till convergence. These methods repeatedly perform complete inference in equation 18 which is a very expensive process especially for large models.
We apply the methodology introduced for paired-dual learning to learn this model. It speeds up training by interleaving updates of θ into dual optimizations over ρ and q. This method enables optimization using partial solutions to inference. The basic idea is iterating small inference update with learning updates; we do not solve inference to completion. First, we improve $\mathbf{z}$ a little bit, next we improve $\mathbf{y}$ slightly, and then we take the gradient with respect to θ. We can stop after a fixed number of iterations or when θ has converged. Learning procedure is shown in algorithm 1. The output of algorithm 1 is the updated weighs. Using these weights we can get the value for latent variables using inference and obtain the belief of unlabeled node.

5 Experiments

5.1 Data set

For our experiments, we use scene understanding dataset [8] used for geometric and semantic scene understanding. The dataset contains 715 images chosen from existing public datasets: LabelMe,

Algorithm 1 Paired-Dual learning on pairwise MRF using Belief Propagation and Bethe entropy

procedure BETHE_PDL($data$) ▷ Create a pairwise MRF and learn the parameters using given data
  $\theta = random(1, n)$ ▷ initializing parameters θ
  while θ has not been converged **do**
    $\tau_q = BP(data, Iter = 1)$ ▷ run BP inference for one step using 2 on partial labeled data
    $H_{Bethe}(\tau_q)$ ▷ Bethe entropy for partial labeled data
    $\tau_\rho = BP(data, Iter = 1)$ ▷ run BP inference for one step using 2 as if data is unlabeled
    compute $H_{Bethe}(\tau_\rho)$ ▷ Bethe entropy as if data is unlabeled
    $\bigtriangledown_\theta = \tau_q - \tau_\rho$ ▷ take the gradient w.r.t θ
    Update θ using $\bigtriangledown_\theta$
  θ ▷ return the updated weights

MSRC, PASCAL VOC and Geometric Context. The selection criteria were for the images to be of outdoor scenes, have approximately 320-by-240 pixels, contain at least one foreground object, and have the horizon position within the image (it need not be visible). Semantic and geometric labels were obtained using Amazon's Mechanical Turk (AMT). The labels are: 0 sky, 1 tree/bush, 2 road/path, 3 grass, 4 water, 5 building, 6 mountain, 7 foreground object. Due to lack of time, we chose a small subset of these images for our training and test. We also scaled down images to 8×10. We randomly select 16 pixels to be latent variable and the task is to learn the model containing latent variables.

5.2 Experiment Setup and Evaluation

In this section, we evaluate the performance of our introduced Bethe paired-dual learning on image segmentation problem. Additionally, we compare Bethe paired-dual learning with subgradient and expectation maximization as baselines regarding convergence and learning speed. To solve optimization problem 11, subgradient method requires two inferences in each optimization step. Expectation maximization, in the other side, is an iterative process of inference over latent variables and subgradient descent to update the parameters. Thus, these two methods have at least two full inferences in each step of updating the parameters. Bethe paired-dual learning, however, solves optimization problem 11 using partial solution to inference. Particularly, we run incomplete inference in each updating step.

We first create Markov random field that includes latent variables. For inference, we use sum-product message passing algorithm to get the marginals (beliefs) of nodes about the labels; but we did not perform full inference; instead, we update message only once. During each outer iteration of each algorithm, we store the current weights and later use these weights to measure the primal objective, 11, and predictive performance on training and test data. In our experiments, for updating the weight parameters, we used L-BFGS-B [1] algorithm which is limited-memory BFGS to handle simple box constraints on variables. It is an iterative method which works by identifying fixed and free variables at every step (using a simple gradient method), and then using the L-BFGS method for the free variables only to get higher accuracy.

In image segmentation task, variable x indicates pixel, y indicates label of the pixel and z is hidden variable which are actually unknown labels for pixels. We set the $L2$ regularizer to 0.1. In belief propagation, we used sum-product message passing where messages are updated synchronously. For evaluation, we keep track of time required for learning optimization problem 11 using our Bethe PDL method and two baselines. We observed that for training 40 scaled down images (8×10), EM optimizes the learning objective in 907.17 seconds, subgradient needs 1112.55 seconds, while Bethe PDL optimizes the objective only in 63 seconds which is a significant improvement. It will be desirable, especially for very large models.

We are also interested to analyze the primal objective trend of our method and two baselines during the optimization process. So, we used the weights stored in the outer iteration of these three algorithms to measure the primal objective. Figure 1 shows the resulting primal objective values. As can be seen, Bethe PDL quickly optimizes the learning objective. It reaches to its optimal value, whereas two baselines are in their very beginning steps of learning.

We plot the training and test accuracy during optimization steps. Figure 2 shows the average accuracy of all training and test data during optimization. Bethe PDL reaches to its highest accuracy earlier

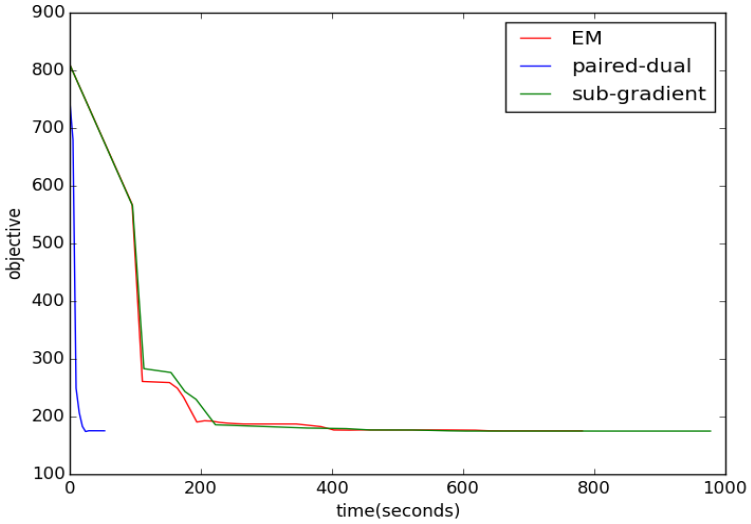
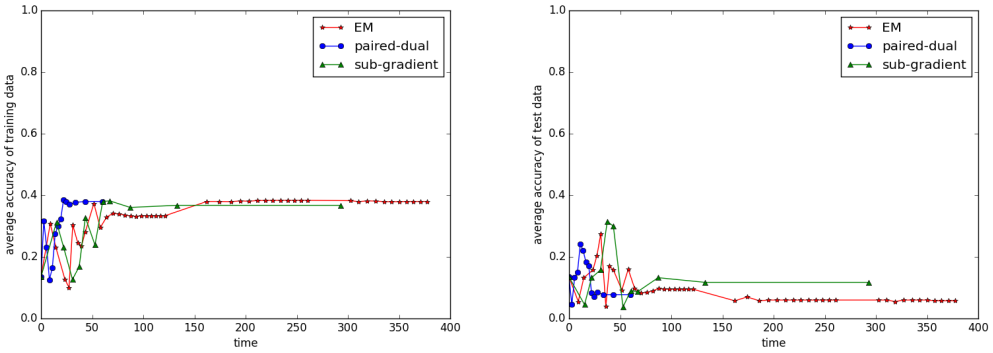


Figure 1: Primal objective score for Bethe PDL, EM, and subgradient in each optimization step. Objective score for Bethe PDL converges much faster.



(a) Average accuracy of training data

(b) Average accuracy of test data

Figure 2: Average accuracy of training and test data for Bethe PDL, EM, and subgradient.

than baselines while we are not observing such behavior for test data. This can happen due to several reasons. We might need more training data, or more interesting and informative features than only (R, G, B) values. Scaling down image would influence the accuracy adversely. We plan to improve the proposed method to gain more accuracy for training as well as test data.

6 Conslusion

In this study, we extend the paired-dual learning framework for discrete Markov random fields that includes latent variables. We substitute the inferences in objective with dual inferences using belief propagation approach. Paired-dual learning was proposed for fast training of latent variable HL-MRFs. Similar to PDL, we evaluate gradients using incomplete dual inference optimization to avoid repeated, full inference. We have demonstrated the effectiveness of our approach on the image segmentation task using pairwise MRF. We show that the time required to learn Bethe PDL model is significantly less than the time needed for EM, and subgradient. Besides, Bethe PDL optimizes the learning objective much faster than these two baselines. We will continue this work by using more complex, informative features while learning with more training samples. We will also include resulting segmented images learnt from our framework.

References

[1] M. Avriel. *Nonlinear Programming: Analysis and Methods*. Dover Books on Computer Science Series. Dover Publications, 2003.

[2] Stephen H Bach, Bert Huang, Jordan Boyd-Graber, and Lise Getoor. Hinge-loss markov random fields: Convex inference for structured prediction. In *Uncertainty in Artificial Intelligence*, 2013.

[3] Stephen H. Bach, Bert Huang, Jordan Boyd-Graber, and Lise Getoor. Paired-dual learning for fast training of latent variable hinge-loss mrfs. In *International Conference on Machine Learning (ICML)*, 2015. Stephen Bach and Bert Huang contributed equally.

[4] Stephen H Bach, Bert Huang, and Lise Getoor. Learning latent groups with hinge-loss markov random fields. In *CML Workshop on Inferning: Interactions between Inference and Learning*, 2013.

[5] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.

[6] P. Chen, F. Chen, and Z. Qian. Road traffic congestion monitoring in social media with hinge-loss markov random fields. 2014.

[7] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *JOURNAL OF THE ROYAL STATISTICAL SOCIETY, SERIES B*, 39(1):1–38, 1977.

[8] S. Gould, R. Fulton, and D. Koller. Decomposing a scene into geometric and semantically consistent regions. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2009.

[9] j. Domke. Structured learning via logistic regression. 2013.

[10] Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques.* The MIT press, 2009.

[11] Yu-Feng Li, Ivor W. Tsang, James T. Kwok, and Zhi-Hua Zhou. Convex and scalable weakly labeled svms. *Journal of Machine Learning Research*, 4514 PAGES = 2151-2188, YEAR = 2013.

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

[13] O. Meshi, D. Sontag, T. Jaakkola, and A. Globerson. Learning efficiently with approximate inference via dual losses. In *International Conference on Machine Learning (ICML)*, 2010.

[14] Judea Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann Publishers Inc, 1998.

[15] Wei Ping, Qiang Liu, and Alexander Ihler. Marginal structured svm with hidden variables. *31th International Conference on MachineLearning, JMLR*, 2014.

[16] H. Poon and P. Domingos. Sum-product networks: A new deep architecture. *In Uncertainty in Artificial Intelligence*, 2011.

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

[18] A. Quattoni, S. Wang, L. P. Morency, M. Collins, and T. Darrell. Hidden-state conditional random fields. *PAMI*, 2007.

[19] Arti Ramesh, Dan Goldwasser, Bert Huang, Hal Daume III, and Lise Getoor. Learning latent engagement patterns of students in online courses. 2014.

[20] A. Schwing, T. Hazan, M. Pollefeys, and R. Urtasun. Efficient structured prediction with latent variables for general graphical models. In *International Conference on Machine Learning (ICML)*, 2012.

[21] B. Taskar, V. Chatalbashev, D. Koller, and C. Guestrin. Learning structured prediction models: A large margin approach. In *Twenty-Second International Conference on Machine Learning (ICML)*, Bonn, Germany, 2005.

[22] L. van der Maaten, M. Welling, and L. Saul. Hidden-unit conditional random fields. *Artificial Intelligence and Statistics*, 2011.

[23] Martin J. Wainwright and Michael I. Jordan. Graphical models, exponential families, and variational inference. *Found. Trends Mach. Learn.*, 1(1-2):1–305, January 2008.

[24] S. B. Wang, A. Quattoni, and L. Morency andD. Demirdjian. Hidden conditional random fields for gesture recognition. *In Proceedings of CVPR*, 2:1521–1527, 2006.

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

[26] Y. Xu, D. Rockmore, and A. Kleinbaum. Hyperlink prediction in hypernetworks using latent social features. *In Discovery Science*, 32, 2013.

[27] Jonathan S Yedidia, William T. Freeman, and Yair Weiss. Generalized belief propagation. In T. K. Leen, T. G. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems 13*, pages 689–695. MIT Press, 2001.

[28] C. Yu and T. Joachims. Learning structural svms with latent variables. *International Conference on Machine Learning*, 2009.

[29] A. Yuille and A. Rangarajan. The concave- convex procedure. *Neural Computation*, 15, 2003.

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