Dynamic Hierarchical Markov Random Fields and their Application to Web Data Extraction

Jun Zhu[*†] JUN-ZHU@MAILS.TSINGHUA.EDU.CN
Zaiqing Nie[‡] ZNIE@MICROSOFT.COM
Bo Zhang[†] DCSZB@MAIL.TSINGHUA.EDU.CN
Ji-Rong Wen[‡] JRWEN@MICROSOFT.COM
[†] Department of Computer Science & Technology, Tsinghua University, Beijing, China
[‡] Web Search & Mining Group, Microsoft Research Asia, Beijing, China

Abstract

Hierarchical models have been extensively studied in various domains. However, existing models assume fixed model structures or incorporate structural uncertainty generatively. In this paper, we propose Dynamic Hierarchical Markov Random Fields (DHMRFs) to incorporate structural uncertainty in a discriminative manner. DHMRFs consist of two parts – structure model and class label model. Both are defined as exponential family distributions. Conditioned on observations, DHMRFs relax the independence assumption as made in directed models. As exact inference is intractable, a variational method is developed to learn parameters and to find the MAP model structure and label assignment. We apply the model to a real-world web data extraction task, which automatically extracts product items for sale on the Web. The results show promise.

1. Introduction

The Web is a vast and rapidly growing repository of information. There are various kinds of objects, such as products, people, and conferences, embedded in webpages. Our recent work on web data extraction (Zhu et al., 2006) introduces an effective template-independent method which makes it possible to use a single extraction model to automatically extract information from all webpages containing the same type of objects. Because of the heterogeneity of webpages, template-independent web object extraction is challenging. Hierarchical models have great advantages in the reduction of extraction error by integrating multi-scale web data extraction tasks (*i.e.* data record detection and attribute labeling), incorporating long distance dependencies, and fusing multi-scale features (Zhu et al., 2006). However, one problem with this method is that the model structure is fixed by pre-constructed vision-trees (here, a vision-tree is a modified HTML tag tree which can represent the visual layout of a webpage better). The fixed structures are not most appropriate for web data extraction. This is because, unaware of semantic labels, it cannot resolve all ambiguities when constructing the model structures (*i.e.* vision-trees). Some closely related nodes may be separated significantly and only connected through a remote ancestor node on the tree. Due to the model's local Markov assumption, it will lose some useful dependencies and result in low accuracy. An extreme case is that the attributes of different objects are intertwined. Fixed hierarchical models are incapable of re-organizing them correctly. This problem has been known as blocky artifact issue in image processing (Irving et al., 1997).

Thus, effective web data extraction models should have the capability to adapt their structures during the inference process. In this paper, we propose an undirected graphical model named *Dynamic Hierarchical Markov Random Fields* (DHMRFs) to achieve the above goal. DHMRFs consist of two parts – structure model and class label model. Both parts are defined as exponential family distributions. Compared to the directed Dynamic Trees (Williams & Adams, 1999) which have been proposed in image processing to address the blocky artifact issue, our model representation is compact and parameter sharing is easy. This is because conditional probability tables (CPTs) are used in Dynamic Trees to represent transition from parent nodes to child nodes. If different CPTs are used for different nodes, it will easily lead to over-parameterization. Thus, layer-wise CPT sharing is always adopted. But in the scenario of web data, sharing CPTs can be difficult because the hierarchical structures are not as regular as the dyadic or quad trees in image processing. Here, different pages can have quite different depths, and nodes from different pages at the same depth can have very diverse semantics. In contrast, DHMRFs define probability distributions via a set of feature functions and

Appearing in *Proceedings of the 24th International Conference on Machine Learning*, Corvallis, OR, 2007.

*The work is done when the author is visiting Microsoft Research Asia.

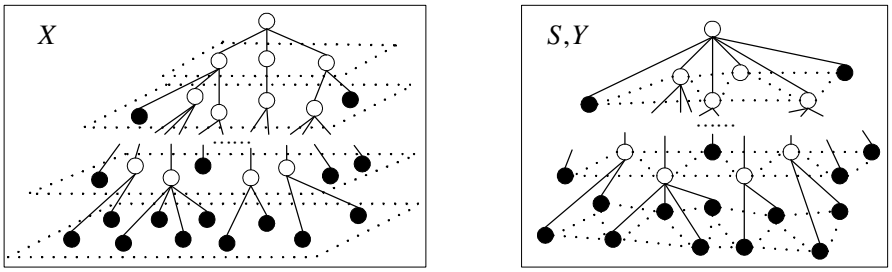

Figure 1. The observations X (left) can be in a hierarchy or other structures, and the right is a *DHMRF* model denoted by S and Y. Nodes are arranged in a layered structure, and vertical edges are selected by posterior probabilities $p(s|x)$. Dotted lines represent the 2D neighborhood system between nodes at the same layer. In both graphs empty nodes are inner nodes and filled nodes are leaf nodes.

weights. These feature functions depend much more on observations and their labels than on the depths of the nodes. Thus, the undirected model is more suitable for diverse web data. Furthermore, as conditional models (Lafferty et al., 2001), DHMRFs relax the independence assumption as made in directed models. Finally, instead of trees in which only parent-child dependencies are assumed, DHMRFs take the triple-wise interactions among neighboring sibling variables and their parent into consideration. These triple-wise dependencies provide more flexibility in encoding useful features.

In undirected dynamic models, parameter estimation is generally intractable, especially when there are hidden variables – both structures and inner variables are hidden in our study. In this paper, a variational algorithm is developed within the paradigm of contrastive divergence mean field learning (Welling & Hinton, 2001) to do parameter estimation and to find the maximum a posterior assignment of labels and the most likely model structures. The performance of our model is demonstrated on a web data extraction task – production information extraction. The results show that our model can achieve high extraction accuracy without tedious manual labeling of inner nodes which is required in the learning of fixed-structured models (Zhu et al., 2006). Note that although we have motivated and evaluated our model only in the field of web data extraction, it could also be applied to other fields since the model itself is general. We leave further examinations as future work.

The rest of the paper is organized as follows. In the next section, we discuss some related hierarchical models. Section 3 describes Dynamic Hierarchical Markov Random Fields, including an approximate inference algorithm. Section 4 provides our empirical evaluation on web data extraction. Section 5 brings this paper to a conclusion, and finally, we give our acknowledgements.

2. Related Work

Multi-scale or hierarchical statistical modeling has shown great promise in image labeling (Kato et al., 1993; Li et al., 2002; He et al., 2004; Kumar & Hebert, 2005), information extraction (Zhu et al., 2006), and human activity recognition (Liao et al., 2005). Based on whether data are observed at multiple scales, two scenarios exist in which hierarchical modeling is appropriate. First, data are observed at different spatial scales and a model is used to integrate information from the different scales. Second, data are observed only at the finest scale and a model is used to induce a particular process at that scale. The introduced intermediate processes or variables can incorporate more complex dependencies to help the target labeling. Another merit of hierarchical models is that they admit more efficient inference algorithms compared to flat models (Willsky, 2002).

Traditional hierarchical models always assume that model structures are fixed or can be constructed via some deterministic methods such as sub-sampling of images (Li et al., 2002), segmentation of webpages (Zhu et al., 2006), and the minimum spanning tree algorithm (Quattoni et al., 2004) with a proper definition of distance. However, in many applications this assumption may not hold. For example, fixed models in image processing often lead to the blocky artifact issue, and similar problems arise in web data extraction due to the diversity of web data. To address this problem some enhanced models have been proposed such as the overlapping tree approach (Ivring et al., 1997). Superior performance is achieved with the improvement of the descriptive component of the model. However, ultimate solutions should deal with the source of the blockiness – fixed model structures. Based on this intuition, Dynamic Trees (Williams et al., 1999) have been proposed, which also consist of two parts – model of structures and model of class labels. However, the key difference between DHMRFs and Dynamic Trees is that DHMRFs are defined as exponential family distributions and thus admit several advantages as in the introduction.

Incorporation of evidence at various scales is examined in a generative manner in (Todorovic & Nechyba, 2005). But our model is discriminative and it can relax the independence assumption among evidence as made in generative models. This is the key idea underlying Conditional Random Fields (Lafferty et al., 2001) which have shown great promise in information extraction (Culotta, et al., 2006; Zhu et al., 2006).

Modeling structural uncertainty has also been studied in relational learning (Getoor et al., 2001). Here, we focus on modeling the structural uncertainty within independently and identically distributed (IID) samples.

Our model is different from Dynamic CRFs (Sutton et al., 2004) which are dynamic in terms of time, that is, they have repetitive model structure and parameters over time, and the structure at each time slice is fixed.

3. Dynamic Hierarchical Markov Random Fields

In this section, we present the detailed description of Dynamic Hierarchical Markov Random Fields. An approximate inference algorithm is developed to do parameter estimation and to find the maximum a posterior model structure and label assignment.

3.1 Model Description

Suppose we are given a set of N vertices, and each vertex is associated with a set of observations. Also suppose the vertices are arranged in a layered manner. Then, hierarchical statistical modeling is a task to construct an appropriate hierarchical model structure and carry out inference about the labels of given observations. Determining the number of layers and the number of nodes at each layer is problem specific. We will give an example of web data extraction in the experiment section. Let S be random variables over hierarchical structures, X be variables over the observations to be labeled, and Y be variables over the corresponding labels. Each component Y_i is assumed to take values from a finite discrete label space $\mathcal{Y}_i$. Here, capitalized characters denote random variables and corresponding lower cases are their instances or configurations, e.g. y is a label assignment and $y_i \in \mathcal{Y}_i$ is one component label. Given observations x, Dynamic Hierarchical Markov Random Fields define a conditional probability distribution $p(\mathrm{s},\mathrm{y}\,|\,\mathrm{x})$ of structure s and label assignment y. An example is shown in Figure 1, where the left graph is observations and the right is an instance of the dynamic model. Applying the chain rule, we get $p(\mathrm{s},\mathrm{y}\,|\,\mathrm{x}) = p(\mathrm{s}\,|\,\mathrm{x})\,p(\mathrm{y}\,|\,\mathrm{s},\mathrm{x})$. Thus, the model consists of two parts – structure model $p(\mathrm{s}\,|\,\mathrm{x})$ and class label model $p(\mathrm{y}\,|\,\mathrm{s},\mathrm{x})$. We explain them as follows:

Structure Model: Let s_{il} be an indicator variable to denote the connectivity between node i and another node l which is at the direct above level. Here, leaf nodes can be at any level except the root node that is taken as a default node for an entire page. For leaf nodes, no child is allowed. We call the parent-child connection *vertical connection*. To retain the computational advantage of tree-structured models, each node is allowed to have only one parent in a particular structure s. To consider the dependencies between neighboring nodes descended from a common parent, *horizontal connection* (*i.e.* connection between nodes at the same level) is incorporated in S. Let n_{ij} be an indicator variable to denote whether node i and node j are adjacent to each other. Here, we assume that the variables n_{ij} are independent of s_{il} and can be determined using some spatial ordering method. This assumption holds in applications such as web data extraction and image processing. As position information is encoded in each node, deterministic spatial ordering can decide the neighborhood system among a set of nodes. In theory, the horizontal neighborhood system can be arbitrary. We consider the 2D cases (Zhu et al., 2005), that is, each node is horizontally connected to all the nearest surrounding nodes in a 2D plane.

Conditioned on observations, the probability distribution of structure model is an exponential family distribution,

$$p(\mathrm{s}\,|\,\mathrm{x}) = \frac{1}{Z_1(\mathrm{x})}\exp\left\{\sum_k \mu_k \sum_{ijl} s_{il}s_{jl}n_{ij}g_k(i,j,l,\mathrm{x})\right\},$$

where a triple (i,j,l) denotes a particular position in the dynamic model. A position can be a time interval in time series or a region of space in random fields. Here, i and j are two nodes at the same layer and l is a node at the direct above layer. $g_k(i,j,l,\mathrm{x})$ are feature functions defined on the three nodes at position (i,j,l), and μ_k are their weights. $Z_1(\mathrm{x})$ is a normalization factor and depends on observations.

Class Label Model: A sample s from the structure model defines a hierarchical Conditional Random Fields (CRFs) (Lafferty et al., 2001). Let α_i^y be an indicator variable to denote the variable Y_i taking the class label y. Then, the conditional probability of a label assignment y is,

$$p(\mathrm{y}\,|\,\mathrm{s},\mathrm{x}) = \frac{1}{Z_2(\mathrm{s},\mathrm{x})}\exp\left\{\sum_k \lambda_k \sum_{ijl} s_{il}s_{jl}n_{ij}\alpha_i^{y_i}\alpha_j^{y_j}\alpha_l^{y_l} f_k(y_i,y_j,y_l,\mathrm{x})\right\},$$

where $f_k(y_i,y_j,y_l,\mathrm{x})$ are feature functions defined on the labels y_i, y_j, and y_l at position (i,j,l), and λ_k are their weights; $Z_2(\mathrm{s},\mathrm{x})$ is a normalization factor and depends on both observations and the given model structure.

Although conditional models take observations as global conditions, when defining feature functions they need to know the "focused observations" at a particular position. For example, in linear-chain CRFs (Lafferty et al., 2001) the observation at time t is among the focused observations when defining feature functions related to label y_t. In general, let t be a position and x_t be the set of focused observations at that position. The mapping function $\zeta : t \mapsto \mathrm{x}_t$ defines the focused observations for each position. In generative models like (Todorovic & Nechyba, 2005), the mapping function is defined to determine the observations generated by the states at a particular position. Moreover, an additional constraint $\forall t \neq s, \mathrm{x}_t \cap \mathrm{x}_s = \Phi$ is also set due to their independence assumption that observations at different positions are conditionally independent given the states at those positions. In conditional models, however, there is no such constraint. The mapping function can be deterministic or stochastic. We assume it to be deterministic in this paper. Now, all feature functions take an additional argument ζ, that is, the feature functions are $g_k(i,j,l,\mathrm{x},\zeta)$ and $f_k(y_i,y_j,y_l,\mathrm{x},\zeta)$.

Now, the joint distribution is also an exponential one,

$$p(\mathrm{s},\mathrm{y}\,|\,\mathrm{x}) = \frac{1}{Z_1(\mathrm{x})Z_2(\mathrm{s},\mathrm{x})}\exp\left\{\begin{array}{l}\sum_k \mu_k \sum_{ijl} s_{il}s_{jl}n_{ij}g_k(i,j,l,\mathrm{x},\zeta) + \\ \sum_k \lambda_k \sum_{ijl} s_{il}s_{jl}n_{ij}\alpha_i^{y_i}\alpha_j^{y_j}\alpha_l^{y_l} f_k(y_i,y_j,y_l,\mathrm{x},\zeta)\end{array}\right\}.$$

In the sequel, we will use $Z(\mathrm{x}) = Z_1(\mathrm{x}) Z_2(\mathrm{s},\mathrm{x})$ to denote the overall normalization factor.

3.2 Parameter Estimation and Labeling

Let $\Theta = \{\mu_1, \mu_2, \cdots, \mu_{K_1}; \lambda_1, \lambda_2, \cdots, \lambda_{K_2}\}$ denote the whole set of the model's parameters. Given a set of training data $D = \{<\mathrm{x}^i, \mathrm{y}_e^i>\}_{i=1}^K$, where x^i is a sample and y_e^i are observed labels. We consider the general case with both hidden hierarchical structure s and hidden labels y_h. For example, in web data extraction only the labels of leaf nodes are observable and both the hierarchical structures and the labels of inner nodes are hidden. So the log-likelihood of the data is incomplete,

$$L(\Theta) = \sum_{i=1:K} \log p(\mathrm{y}_e^i \mid \mathrm{x}^i) = \sum_{i=1:K} \log\left(\sum_{\mathrm{s},\mathrm{y}_h} p(\mathrm{s},\mathrm{y}_h,\mathrm{y}_e^i \mid \mathrm{x}^i) \right).$$

This function does not have a closed form solution because of the marginalization taking place within logarithm. In the following, we derive an upper bound of the negative log-likelihood. Then, contrastive divergence learning (Hinton, 2002) is applied as an approximation.

Let $q(\mathrm{s},\mathrm{y}_h \mid \mathrm{y}_e,\mathrm{x})$ be an approximation of the distribution $p(\mathrm{s},\mathrm{y}_h \mid \mathrm{y}_e,\mathrm{x})$. With a little abuse of notations, we will use $q(\mathrm{s},\mathrm{y}_h)$ to denote $q(\mathrm{s},\mathrm{y}_h \mid \mathrm{y}_e,\mathrm{x})$. We also ignore the summation operator in the log-likelihood during the following derivations as there is no essential difference between one sample and a set of independently and identically distributed (IID) samples. The optimal approximation is the distribution that has the minimum Kullback-Leibler divergence between $q(\mathrm{s},\mathrm{y}_h)$ and $p(\mathrm{s},\mathrm{y}_h \mid \mathrm{y}_e,\mathrm{x})$. The KL divergence is defined as

$$KL(q \,\|\, p) = \sum_{\mathrm{s},\mathrm{y}_h} q(\mathrm{s},\mathrm{y}_h) \log \frac{q(\mathrm{s},\mathrm{y}_h)}{p(\mathrm{s},\mathrm{y}_h \mid \mathrm{y}_e,\mathrm{x})}.$$

Take $p(\mathrm{s},\mathrm{y}_h \mid \mathrm{y}_e,\mathrm{x}) = p(\mathrm{s},\mathrm{y}_h,\mathrm{y}_e \mid \mathrm{x}) / p(\mathrm{y}_e \mid \mathrm{x})$ into the above equation and use the non-negativity of KL divergence, we can easily derive an upper bound of the negative log-likelihood $-L(\Theta) = -\log p(\mathrm{y}_e \mid \mathrm{x})$, that is,

$$L(\Theta) = \sum_{\mathrm{s},\mathrm{y}_h} q(\mathrm{s},\mathrm{y}_h)\left[\log q(\mathrm{s},\mathrm{y}_h) - \log p(\mathrm{s},\mathrm{y}_h,\mathrm{y}_e \mid \mathrm{x})\right] \geq -L(\Theta).$$

By analogy with statistical physics, the upper bound, which is actually a KL divergence, can be expressed as the difference of two free energies: $L(\Theta) = F_0 - F_\infty$, where the first term is the free energy when we use data distribution with observable labels clamped to their values, and the second $F_\infty = -\log Z(\mathrm{x})$ is the free energy when we use model distribution with all variables free.

Now, the problem is to optimize the upper bound. The derivatives of $L(\Theta)$ with respect to λ_k are,

$$\begin{aligned}
\frac{\partial L(\Theta)}{\partial \lambda_k} &= \frac{\partial}{\partial \lambda_k} \left\langle -\log p(\mathrm{s},\mathrm{y}_h,\mathrm{y}_e \mid \mathrm{x}) \right\rangle_{q(\mathrm{s},\mathrm{y}_h)} \\
&= -\sum_{ijl} \left\langle s_{il} s_{jl} n_{ij} \right\rangle_{q(\mathrm{s},\mathrm{y}_h)} \sum_{y_l y_j y_l} \left\langle \alpha_i^{y_i} \alpha_j^{y_j} \alpha_l^{y_l} \right\rangle_{q(\mathrm{s},\mathrm{y}_h)} f_k(y_i, y_j, y_l, \mathrm{x}, \zeta) - \frac{\partial F_\infty}{\partial \lambda_k} \\
&= -\sum_{ijl} n_{ij} \left\langle s_{il} s_{jl} \right\rangle_{q(\mathrm{s},\mathrm{y}_h)} \sum_{y_l y_j y_l} \left\langle \alpha_i^{y_i} \alpha_j^{y_j} \alpha_l^{y_l} \right\rangle_{q(\mathrm{s},\mathrm{y}_h)} f_k(y_i, y_j, y_l, \mathrm{x}, \zeta) - \frac{\partial F_\infty}{\partial \lambda_k}
\end{aligned} \quad (1),$$

where $\langle . \rangle_p$ is the expectation under the distribution p. The last equation holds because of the assumption that the neighborhood system between sibling nodes is determined independent of their parents.

Similarly, the derivatives with respect to μ_k are,

$$\frac{\partial L(\Theta)}{\partial \mu_k} = -\sum_{ijl} n_{ij} \left\langle s_{il} s_{jl} \right\rangle_{q(\mathrm{s},\mathrm{y}_h)} g_k(i, j, l, \mathrm{x}, \zeta) - \frac{\partial F_\infty}{\partial \mu_k} \quad (2).$$

In (1) and (2), the derivatives of the equilibrium free energy F_∞ are essentially intractable in the case of Dynamic Hierarchical Markov Random Fields. However, by viewing the equilibrium distribution as the distribution of a Markov chain at time $t = \infty$ starting with data distribution, Markov chain Monte Carlo (MCMC) method can be used to reconstruct an approximation distribution $q_i(\mathrm{s},\mathrm{y}_h,\mathrm{y}_e)$ within several steps. This is the basic idea of contrastive divergence learning (Hinton, 2002). Now, the upper bound is approximated by,

$$\begin{aligned}
L(\Theta) &= F_0 - F_\infty \\
&\approx F_0 - F_i = KL(q_0 \,\|\, p) - KL(q_i \,\|\, p) \equiv CF_i^{APP},
\end{aligned}$$

where $q_0 = q(\mathrm{s},\mathrm{y}_h)$ is optimized with observable labels clamped to their values, and $q_i(\mathrm{s},\mathrm{y}_h,\mathrm{y}_e)$ is optimized with all variables free starting with q_0. As shown in (Hinton, 2002), CF_i^{APP}, known as contrastive divergence, is non-negative. Some analyses of contrastive divergence learning appear in (Yuille, 2004; Carreira-Perpinan & Hinton, 2005). In the sequel, we will set $i = 1$.

Now, the derivatives of CF_1^{APP} with respect to the model's parameters are as in (1) and (2) but with the derivatives of F_∞ replaced by,

$-\sum_{ijl} n_{ij} \left\langle s_{il} s_{jl} \right\rangle_{q_1(\mathrm{s},\mathrm{y}_h,\mathrm{y}_e)} \sum_{y_l y_j y_l} \left\langle \alpha_i^{y_i} \alpha_j^{y_j} \alpha_l^{y_l} \right\rangle_{q_1(\mathrm{s},\mathrm{y}_h,\mathrm{y}_e)} f_k(y_i, y_j, y_l, \mathrm{x}, \zeta)$,

and $-\sum_{ijl} n_{ij} \left\langle s_{il} s_{jl} \right\rangle_{q_1(\mathrm{s},\mathrm{y}_h,\mathrm{y}_e)} g_k(i, j, l, \mathrm{x}, \zeta)$ respectively.

Generally, stochastic sampling is quite time demanding in constructing q_1. In contrast, the deterministic mean field variant (Welling & Hinton, 2001) is more efficient. The learning procedure consists of two phases – wake phase and sleep phase. Wake phase is to optimize q_0 and sleep phase is to optimize q_1. We address the wake phase first.

Assume the variational distribution can be factorized as $q_0 = q(\mathrm{s},\mathrm{y}_h) = q(\mathrm{s}) q(\mathrm{y}_h)$, and we get,

$$KL(q_0 \,\|\, p) = -\left\langle \log p(\mathrm{s},\mathrm{y}_h,\mathrm{y}_e \mid \mathrm{x}) \right\rangle_{q(\mathrm{s},\mathrm{y}_h)} - H(q(\mathrm{s})) - H(q(\mathrm{y}_h)) \quad (3),$$

where $H(p) = -\langle \log p \rangle_p$ is the entropy of distribution p. To efficiently optimize q_0, more assumptions need to be made about the family of distributions of $q(\mathrm{s})$ and $q(\mathrm{y}_h)$. Here, we adopt the naïve mean field approximation. The basic idea underlying mean field theory (Jordan et al., 1999) is to make a distribution a factorized one by introducing additional independence assumptions. This factorized distribution leads to computational tractability.

The simplest naïve mean field is to assume that interacted variables are independent and the joint distribution is a

product of single variable marginal probabilities. Let μ_{il} be the probability of node i being connected to node l, and m_i^y be the probability of variable Y_i being at state y. As we assume variables n_{ij} are determined independent of s_{il}, the mean field distributions[1] are,

$$q(\mathrm{s}) = \prod_{il} [\mu_{il}]^{s_{il}} \text{ and } q(\mathrm{y}_h) = \prod_{iy} \left[m_i^y \right]^{\alpha_i^y}.$$

Substitute the above distributions into (3) and keep $q(\mathrm{y}_h)$ fixed, then we get

$$KL(q_0 \| p) = -\left\langle \log p(\mathrm{s}, \mathrm{y}_h, \mathrm{y}_e \mid \mathrm{x}) \right\rangle_{q(\mathrm{s},\mathrm{y}_h)} - H(q(\mathrm{s})) + c,$$

where c is a constant. Let the derivative over μ_{il} equal zero, and we get $\log \mu_{il} = s_{il} \left\langle \log p(\mathrm{s}, \mathrm{y}_h, \mathrm{y}_e \mid \mathrm{x}) \right\rangle_{q(\mathrm{y}_h)} + const$. Thus,

$$\mu_{il} \propto \exp\left\{ \begin{array}{l} \sum_k \mu_k s_{il} \sum_j \left\langle s_{jl} \right\rangle_{q(s)} n_{ij} g_k(i,j,l,\mathrm{x}) + \\ \sum_k \lambda_k s_{il} \sum_j \left\langle s_{jl} \right\rangle_{q(s)} n_{ij} \sum_{y_1 y_2 y_3} \left\langle \alpha_i^{y_1} \alpha_j^{y_2} \alpha_l^{y_3} \right\rangle_{q(\mathrm{y}_h)} f_k(y_1, y_2, y_3, \mathrm{x}, \zeta) \end{array} \right\} \quad (4).$$

Normalization will lead to the desired probabilities μ_{il}.

Similarly, keep $q(\mathrm{s})$ fixed and we get

$$KL(q_0 \| p) = -\left\langle \log p(\mathrm{s}, \mathrm{y}_h, \mathrm{y}_e \mid \mathrm{x}) \right\rangle_{q(\mathrm{s},\mathrm{y}_h)} - H(q(\mathrm{y}_h)) + c',$$

where c' is another constant. Let the derivative over m_i^y equal zero, and we get

$$m_i^y \propto \exp \sum_k \lambda_k \sum_{jly_1y_2} \left\{ \begin{array}{l} n_{ij} \left\langle s_{il} s_{jl} \right\rangle_{q(\mathrm{s})} \left\langle \alpha_j^{y_1} \alpha_l^{y_2} \right\rangle_{q(\mathrm{y}_h)} f_k(y, y_1, y_2, \mathrm{x}, \zeta) \\ + n_{ij} \left\langle s_{jl} s_{il} \right\rangle_{q(\mathrm{s})} \left\langle \alpha_j^{y_1} \alpha_l^{y_2} \right\rangle_{q(\mathrm{y}_h)} f_k(y_1, y, y_2, \mathrm{x}, \zeta) \\ + n_{jl} \left\langle s_{jl} s_{li} \right\rangle_{q(\mathrm{s})} \left\langle \alpha_j^{y_1} \alpha_i^{y_2} \right\rangle_{q(\mathrm{y}_h)} f_k(y_1, y_2, y, \mathrm{x}, \zeta) \end{array} \right\} \quad (5).$$

Equations (4) and (5) are a set of coupled equations, also known as mean field equations. These equations are iteratively solved for a fixed point solution. Intuitively, parameters μ_{il} are updated by expected contributions from possible parents and neighbors, and similar for m_i^y. In (4) and (5), structure parameters μ_{il} depend on class label assignments, and m_i^y depend on expected structure connectivity. Thus, model structure selection is integrated with label assignment during the inference.

Now, we have presented a mean field approximation of the wake phase. To finish the sleep phase, the same mean field equations are enforced by coordinate descent alternating between observable variables Y_e and hidden variables S and Y_h. When first optimizing (5) for Y_e, the initial distribution of hidden variables are set as the optimal distribution at the end of wake phase. Then, take the optimal distribution of the former step as initial distribution of Y_e and optimize (4) and (5) to get an approximate distribution of hidden variables. For wake phase, initial distributions can be random and convergence is arrived. But for sleep phase, a few steps are required to guarantee the improvement of CF_1^{APP}.

Thus, all the terms in (1), (2), (4), and (5) can be calculated. The whole parameter estimation algorithm is as follows. First apply (4) and (5) to iteratively compute the marginal probabilities of both wake and sleep phases, and CF_1^{APP} and its derivatives with respect to model parameters are calculated. Then, gradient-based optimization algorithms are applied to update model parameters. Here, we use the limited memory quasi-Newton method (Liu & Nocedal, 1989). The learning procedure is iterated until the relative change of CF_1^{APP} is below some threshold. Although no guarantee exists that global optimization will be achieved, empirical studies show that this algorithm performs well.

As for labeling, when a testing example comes in, equations (4) and (5) are iteratively solved with all variables hidden for a fixed point solution. At the end of convergence, the maximum a posterior model structure can be constructed from the probabilities μ_{il}, and the most likely label assignments can be found from the marginal probabilities m_i^y.

4. Experiments

In this section, we evaluate DHMRFs on a real-world web data extraction task – production information extraction. We compare our model with Hierarchical Conditional Random Fields (HCRFs) (Zhu et al., 2006), Dynamic Trees (Williams et al., 1999), and fixed tree models. The results demonstrate the merits of our model. Empirical studies about the inference algorithm are also presented.

4.1 Datasets and Methods

Web data extraction is an information extraction (IE) task that identifies information of interest from webpages, and production information extraction is a web data extraction task that identifies product items for sale on the web. For each product item, four attributes – *Name*, *Image*, *Price*, and *Description* are extracted in our experiments. The difference of web data extraction from traditional IE is that various types of structural dependencies between the HTML elements exist, e.g. the HTML tag tree is itself hierarchical. Extending statistical models to handle these structural dependencies has received great attention of late. In this paper, we address the limitations of the fixed-structured hierarchical model (Zhu et al., 2006). To compare with that fixed-structured hierarchical model, we use the same datasets as (Zhu et al., 2006). The datasets consist of both list and detail pages. A list page contains several structured data records while a detail page contains only detailed information about a single record. Examples of list and detail pages are illustrated in (Zhu et al., 2006). The dataset *LDST* contains 771 list pages and

[1] Let s_v denote the joint variable of vertical connection s_{il} and s_h denote the joint variable of horizontal connection n_{ij}, then $q(\mathrm{s}) = q(\mathrm{s}_v, \mathrm{s}_h) = q(\mathrm{s}_v) q(\mathrm{s}_h \mid \mathrm{s}_v)$. Based on the assumption that s_h is independent of s_v, $q(\mathrm{s}_h \mid \mathrm{s}_v)$ is an indicator function, and takes all the probability one if only if s_h is the allowed structure.

Table 1. Performance of different models on production information extraction. Here, "Desc" denotes the attribute Description.

Data Sets		LDST				DDST			
Models		*F-Trees*	*D-Trees*	*HCRFs*	*DHMRFs*	*F-Trees*	*D-Trees*	*HCRFs*	*DHMRFs*
P	Name	0.890	0.879	0.911	**0.952**	0.829	0.785	0.835	**0.874**
	Image	0.959	0.951	0.966	**0.988**	0.972	0.928	**0.978**	**0.978**
	Price	0.960	0.937	0.963	**0.978**	0.976	0.947	0.986	**0.989**
	Desc	0.804	0.800	0.788	**0.828**	0.722	0.698	0.663	**0.730**
R	Name	0.842	0.744	0.882	**0.928**	0.779	0.684	0.761	**0.799**
	Image	0.908	0.805	0.936	**0.958**	0.868	0.809	0.892	**0.898**
	Price	0.910	0.794	0.936	**0.949**	0.888	0.826	0.899	**0.905**
	Desc	0.762	0.678	0.764	**0.811**	0.641	0.609	0.604	**0.668**
F1	Name	0.865	0.806	0.896	**0.940**	0.803	0.731	0.796	**0.835**
	Image	0.933	0.872	0.951	**0.973**	0.917	0.864	0.933	**0.936**
	Price	0.934	0.860	0.948	**0.963**	0.930	0.882	0.940	**0.945**
	Desc	0.782	0.734	0.776	**0.819**	0.679	0.650	0.632	**0.698**
Avg_F1		0.879	0.818	0.893	**0.924**	0.832	0.782	0.825	**0.854**
Blk_IA		0.869	0.837	0.890	**0.940**	0.809	0.762	0.817	**0.853**

the dataset *DDST* contains 450 detailed pages. Among all these pages, 200 list pages and 150 detail pages are used as training data in (Zhu et al., 2006). We use the same setting for training and testing all the models.

We compare our model with *HCRFs*, Dynamic Trees (*D-Trees*), and fixed-structured tree models (*F-Trees*). For *HCRFs* and *F-Trees*, all training pages are hierarchically labeled with leaf labels and inner labels as defined in (Zhu et al., 2006). The training is complete, and exact message passing algorithms are used to learn their parameters and find MAP label assignments. For *DHMRFs* and *D-Trees*, labels of leaf nodes are kept the same and inner labels are hidden during learning. For the incomplete training, we apply the variational method developed in this paper for *DHMRFs*. Mean field approximation is also used for Dynamic Trees. For *DHMRFs* and *HCRFs*, the same set of feature functions are used for class label assignment. Details about the definition of these feature functions are presented in (Zhu et al., 2006).

To apply the dynamic models *DHMRFs* and *D-Trees*, initial configuration of the model structure must be carried out first. Basically, we need to initially set the number of layers and the number of nodes at each layer. It may be different for different application domains to set the initial configuration. For image processing, it can be done via sub-sampling or wavelet filtering. For web data extraction, the data are represented as texts, images, buttons, and so on. These atomic information units are more expressive compared to image pixels. There is definitely no benefit to view a webpage as a collection of image pixels and then apply the methods in image processing. Here, we use the same number of layers (and the same number of nodes at each layer) in dynamic models as in the fixed vision-trees (Zhu et al., 2006).

For *D-Trees*, two sets of parameters – conditional probability tables (CPTs) and affinities, need to be set. We keep the affinities fixed and learn the model's CPTs. To avoid over-parameterization, layer-wise CPT sharing is adopted in previous work. However, for heterogeneous web data, three-layer-wise sharing is better. That is, every three layers from the top down share one CPT. To incorporate evidence, we use the class-independent model (Storkey et al., 2003) with emission distributions set as the empirical frequencies in the training dataset. CPTs are also initialized as frequencies. To avoid zero probabilities of unseen samples, Laplace's rule is used with pseudocount set at one. Our study shows that when the affinities are set as 0 for the natural parent, -1 for the nearest neighbors of the natural parent, and -3 for the null parent, better performance is achieved compared with previously used settings. The CPTs used for our experiments are achieved with 10 iterations.

4.2 Results and Discussions

4.2.1 Extraction Accuracy

Table 1 shows the extraction accuracy of different models. We use the standard measures *P*recision, *R*ecall, and their harmonic mean *F1* value. Two comprehensive measures Average F1 (Avg_F1) and Block Instance Accuracy (Blk_IA) (Zhu et al., 2005) are also used. Block Instance Accuracy is the percentage of records whose *Name*, *Image*, and *Price* are all correctly labeled. Note that each product can have only one price which is the current price for sale. Other prices detected are treated as errors.

From the results, we can see that *DHMRFs* achieve the highest performance on both datasets. Compared to the fixed *HCRFs*, on *LDST* about 3 points in Average F1 and about 5 points in Block Instance Accuracy are gained. For *Name* and *Description*, more than 4 points are achieved in both precision and recall, and for *Image* and *Price* the improvements are slightly smaller (about 2 points in F1). This is because *Image* and *Price* are usually more distinctive than the other attributes. So both models perform quite well. On *DDST*, the improvements in *Name* are about 4 points in both precision and recall, and for *Description* the improvements are about 7 points in both

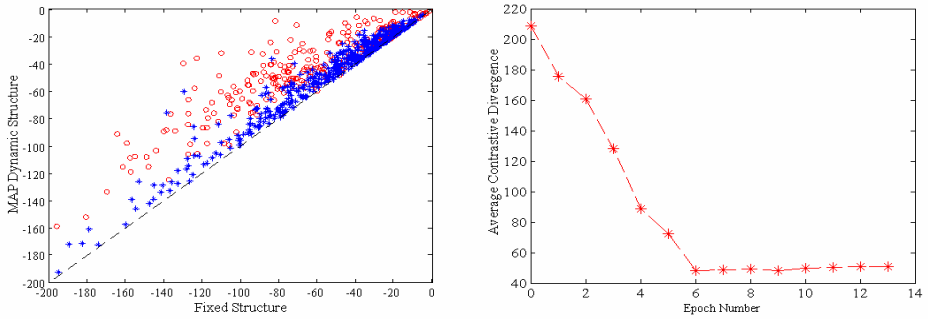

Figure 2. Plot (a) shows the log posteriors of MAP dynamic structures against those of fixed structures. Samples in asterisks are from *LDST* and those in circles are from *DDST*. Plot (b) is the change of average contrastive divergence with respect to iteration numbers.

precision and recall. Small improvements are achieved in *Image* and *Price* due to the same reason as in list pages.

The improvements demonstrate the merits of *DHMRFs*. First, *DHMRFs* can incorporate the two-dimensional neighborhood dependencies among the nodes at the same level, which have been shown to be useful in (Zhu et al., 2005), while *HCRFs* must take a sequentialization to put these nodes into a chain. By dynamically selecting connections between different nodes, *DHMRFs* can bring together the attributes of the same object (here, an object is a product item), and thus the correlation between these attributes can be strengthened. Second, *DHMRFs* can deal with webpages with intertwined attributes (Zhai & Liu, 2005). For these webpages, the attributes of different objects are intertwined in HTML tag trees. Unaware of semantic labels, the constructed vision-trees (Zhu et al., 2006) also have intertwined attributes. In these cases, the fixed-structured *HCRFs* cannot correctly detect data records by simply assigning labels to the nodes of a vision-tree. Instead, as structure selection is integrated with labeling in *DHMRFs*, the dynamic model can properly group the attributes of the same object and at the same time separate the attributes of different objects with the help of semantic labels. The semantic labels have been shown helpful in detecting data records (*i.e.* groups of attributes) in (Zhu et al., 2006). Note that although intertwined cases are usually fewer than non-intertwined cases, they are not sparse samples in our model. This is because although their edge connections in HTML tag trees are somewhat different from non-intertwined ones, the visual features they share are almost the same. Thus, training samples with or without intertwined cases can teach a good model.

Compared to the fixed *F-Trees*, the worse performance of *D-Trees* is quite counter-intuitive. However, a close examination of the results reveals that the reason for the worse performance is due to the less discriminative power of *D-Trees*. As we have stated, for diverse web data CPT sharing can be difficult. Although empirical studies can find a good sharing method, we couldn't learn an optimal model with a limited set of training samples. Furthermore, its generative characteristic causes difficulty in encoding useful features. In this way, more uncertainty in structure selection couldn't be resolved than that in *DHMRFs*. This is evident if we look at the average log-likelihood of the MAP connections over all samples and all nodes. For *D-Trees* the average value is -0.4080, and for *DHMRFs* it is -0.3170. In terms of probability, they are equivalent to 0.6650 and 0.7283 respectively. The less discriminative power of *D-Trees* causes additional errors in constructing model structures even for the non-intertwined cases, and thus hurts the accuracy of record detection and attribute labeling. So, *D-Trees* perform worse than *F-Trees*, which can deal with the non-intertwined cases well. The results also show that the directed tree models can perform well on our datasets, but are inferior to *HCRFs*.

4.2.2 Fitness of Model Structure

Figure 2(a) compares the posterior probabilities of the MAP structures constructed by *DHMRFs* with those of the fixed structures. In terms of the number of nodes, the sizes of webpages change from 39 to 576 (average 166) in *LDST*, and the log posteriors change from -503.80 to -4.49 (average -50.7). In *DDST*, sizes range from 14 to 705 (average 131), and log posteriors range from -184.40 to -1.72 (average -42.47). Here, we only present the samples whose log posteriors are between -200 and 0 because most of the samples (>97%) fall into this interval. We can see that the MAP structures by *DHMRFs* always appear above the equal probability line. Thus, the structures found by the dynamic model have higher posterior probabilities. Another observation is that the distribution of samples from *DDST* is more disperse than that of the samples from *LDST*. The reason is that in list pages the attributes of an object always concentrate into small clusters while they can scatter anywhere in detail pages.

4.2.3 Study about the Inference Algorithm

Figure 2(b) shows the change of average contrastive divergence with respect to iteration numbers in the learning of *DHMRFs*. To initialize the algorithm, at the wake phrase m_i^y are set to a uniform distribution plus a Gaussian noise with zero mean and variance 0.01, and μ_{il} are set to a random distribution. The model weights are initialized to zero. We can see that before 7 iterations average contrastive divergence decreases stably. And after 7, slight disturbances appear. But as for extraction accuracy, marginal changes occur (no more than 0.5 point in Block Instance Accuracy). Thus, the learning algorithm

is quite stable. All the above results are achieved at iteration 7. The same initialization is used in labeling, and by running both learning and labeling many times, we observe that the algorithm is insensitive to the random initialization. Since the mean field equations are locally calculated and their update can typically converge within 5 iterations, both the learning and labeling are efficient.

5. Conclusions

In this paper, we propose Dynamic Hierarchical Markov Random Fields to discriminatively incorporate structural uncertainty in hierarchical modeling. By dynamically selecting connections between variables, it can address the blocky artifact issue in diverse web data extraction. Compared to directed models, DHMRFs are compact in representation and powerful in encoding useful features. The model admits efficient variational approximation algorithms to learn parameters and to do labeling. We apply the proposed model to web data extraction. The results demonstrate great promise, and show that it is possible to alleviate the burden of manual labeling of inner nodes in learning fixed-structured models.

Acknowledgments

We are grateful to the anonymous reviewers for their valuable comments to help improve the presentation of this paper. The authors Jun Zhu and Bo Zhang are supported by National Natural Science Foundation of China, Grant No.60621062, and National Key Foundation R&D Project, Grant No.2003CB317007, 2004CB318108.

References

Carreira-Perpinan, M. A., & Hinton, G. E. (2005). On contrastive divergence learning. *Artificial Intelligence and Statistics*.

Culotta, A., Kristjansson, T., McCallum, A., & Viola, P. (2006). Corrective feedback and persistent learning for information extraction. *Artificial Intelligence Journal*.

Getoor, L., Friedman, N., Koller, D., & Taskar, B. (2001). Probabilistic models of relational structure. *Proc. of ICML*.

He, X., Zemel, R. S., & Carreira-Perpinan, M. A. (2004). Multiscale conditional random fields for image labeling. *Proc. of CVPR*.

Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural Computation*.

Irving, W. W., Fieguth, P. W., & Willsky, A. S. (1997). An overlapping tree approach to multiscale stochastic modeling and estimation. *IEEE Trans. on Image Processing*.

Jordan, M., Ghahramani, Z., Jaakkola, T., & Saul, L. (1999). An introduction to variational methods for graphical models. *Learning in Graphical Models*.

Kato, Z., Berthod, M., & Zerubia, J. (1993). Multiscale Markov random field models for parallel image classification. *Proc. of ICCV*.

Kumar, S., & Hebert, M. (2005). A hierarchical field framework for unified context-based classification. *Proc. of ICCV*.

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

Li, J., Gray, R. M., & Olshen, R. A. (2000). Multiresolution image classification by hierarchical modeling with two-dimensional hidden Markov models. *IEEE Trans on Information Theory*.

Liao, L., Fox, D., & Kautz, H. (2005). Location-based activity recognition. *Proc. of NIPS*.

Liu, D. C., & Nocedal, J. (1989). On the limited memory BFGS method for large scale optimization. *Mathematical Programming* 45, pp. 503-528.

Quattoni, A., Collins, M., & Darrell, T. (2004). Conditional random fields for object recognition. *Proc. of NIPS*.

Storkey, A. J. (2000). Dynamic trees: a structured variational method giving efficient propagation rules. *Proc. of UAI*.

Storkey, A. J., & Williams, C. K. I. (2003). Image modeling with position-encoding dynamic trees. *IEEE Trans. on Pattern Analysis and Machine Intelligence*.

Sutton, C., Rohanimanesh, K., & McCallum, A. (2004). Dynamic conditional random fields: factorized probabilistic models for labeling and segmenting sequence data. *Proc. of ICML*.

Todorovic, S., & Nechyba, M. C. (2005). Dynamic trees for unsupervised segmentation and matching of image regions. *IEEE Trans. on Pattern Analysis and Machine Intelligence*.

Welling, M., & Hinton, G. E. (2001). A new learning algorithm for mean field Boltzmann machines. *Proc. of ICANN*.

Williams, C. K. I., & Adams, N. J. (1999). DTs: dynamic trees. *Proc. of NIPS*.

Willsky, A. S. (2002). Multiresolution Markov models for signal and image Processing. *Proc. of the IEEE*.

Winn, J., & Bishop, C. M. (2005). Variational message passing. *Journal of Machine Learning Research*.

Yuille, A. (2004). The convergence of contrastive divergence. *Proc. of NIPS*.

Zhai, Y., & Liu, B. (2005). Web data extraction based on partial tree alignment. *Proc. of WWW*.

Zhu, J., Nie, Z., Wen, J-R., Zhang, B., & Ma, W-Y. (2005). 2D conditional random fields for web information extraction. *Proc. of ICML*.

Zhu, J., Nie, Z, Wen, J-R., Zhang, B., & Ma, W-Y. (2006). Simultaneous record detection and attribute labeling in web data extraction. *Proc. of SIGKDD*.