Incremental Bayesian Networks for Structure Prediction

Ivan Titov IVAN.TITOV@CUI.UNIGE.CH
University of Geneva, 24 rue General Dufour, CH-1211 Geneva, Switzerland

James Henderson JAMES.HENDERSON@ED.AC.UK
University of Edinburgh, 2 Buccleuch Place, Edinburgh EH8 9LW, United Kingdom

Abstract

We propose a class of graphical models appropriate for structure prediction problems where the model structure is a function of the output structure. Incremental Sigmoid Belief Networks (ISBNs) avoid the need to sum over the possible model structures by using directed arcs and incrementally specifying the model structure. Exact inference in such directed models is not tractable, but we derive two efficient approximations based on mean field methods, which prove effective in artificial experiments. We then demonstrate their effectiveness on a benchmark natural language parsing task, where they achieve state-of-the-art accuracy. Also, the model which is a closer approximation to an ISBN has better parsing accuracy, suggesting that ISBNs are an appropriate abstract model of structure prediction tasks.

1. Introduction

In recent years, structure prediction problems, i.e. classification problems with a large (or infinite) structured set of output categories, have attracted much attention. These problems frequently arise in natural language processing (e.g. prediction of phrase structure trees for sentences), biology (e.g. protein structure prediction), chemistry, or image processing. To build probabilistic models of such problems, it is common to decompose the output structures (e.g. phrase structure trees, protein structures) into a sequence of decisions about the output. We can then construct a history-based probability model for these sequences:

$$P(S) = P(D^1, ..., D^m) = \prod_t P(D^t | D^1, \ldots, D^{t-1}), \quad (1)$$

where S is the output structure and $D^1, \ldots, D^m$ is its equivalent sequence of decisions.

We would like to build a graphical model of this decision sequence which allows us to infer each of these conditional probabilities. One approach would be to use a dynamic graphical model with latent state variables (Murphy, 2002), but the resulting graphical model would only have arcs which are local in the decision sequence. Many problems have underlying statistical dependencies which are local only in their output structure, not in any possible decision sequence. For example, in natural language sentences, subject-verb agreement can be expressed via a specific structural configuration in the phrase tree, but the subject and verb may be arbitrarily far apart in the decision sequence. To build graphical models for such non-Markovian problems, we need the arcs of the graphical model to be dependent on the output structure.

The most common approach to building probability models for such problems is to simply not have any latent variables (e.g. (Charniak, 2000; Collins, 1999; Durbin et al., 2003)), but this relies on a hand-built set of features to represent the unbounded decision histories in (1). One alternative proposal (Henderson, 2003) was a model which used the hidden units of a neural network to induce a set of history features. This model achieved state-of-the-art results because its pattern of interconnection between hidden layers was defined in terms of locality in the output structure, as argued for above. However, there was no clear probabilistic semantics for the induced hidden representations.

In this paper we propose a class of graphical models which we call Incremental Sigmoid Belief Networks (ISBNs), which are closely related to the neural network of (Henderson, 2003), but which have a clear

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

probabilistic semantics for all their variables. ISBNs are a kind of Sigmoid Belief Network (Neal, 1992), but are dynamic models and have an incrementally specified model structure. Each position in the decision sequence has a vector of latent state variables, which are connected to variables from previous positions via a pattern of arcs determined by the previous decisions. This gives us a form of switching model (Murphy, 2002), where each decision switches the model structure used for the remaining decisions. In other words, the model structure is specified incrementally by the decision sequence. Because these models use directional arcs and only allow decisions to switch the future model structure, the portion of the model structure which effects the inference of any given $P(D^t|D^1,\ldots,D^{t-1})$ is always known, thereby avoiding the need to sum over model structures, as discussed in section 3. As we will show in this paper, these properties of ISBNs allow us to have large numbers of latent state variables without making the models impractical to use.

Large numbers of latent variables in heavily interconnected directed models make exact inference intractable. We demonstrate the practical applicability of these models by providing efficient approximations. We consider two forms of approximation for ISBNs, a feed-forward neural network approximation (NN) and a form of mean field approximation (MF) (Saul & Jordan, 1999). We first show that the neural network model in (Henderson, 2003) can be viewed as a coarse approximation to inference with ISBNs. We then propose an incremental mean field method, which provides an improved approximation but remains tractable. Both these approximations give us valid probability models.

We performed two empirical evaluations. In the first experiment, we trained both of the approximation models on artificial data generated from random ISBNs. The NN model achieves a 60% average relative error reduction over a baseline model and the MF model achieves a further 27% average relative error reduction over the NN model. These results demonstrate that the distribution of output structures specified by an ISBN can be approximated, that these approximations can be learned from data, and that the MF approximation is indeed better than the NN approximation. In the second experiment, we apply both of the approximation models to phrase structure parsing with data from the Wall Street Journal Penn Treebank. The MF model achieves statistically significant error reduction of about 8% over the NN model. Results of the MF model are non-significantly worse (less than 1% relative error increase) than the results of the best history-based model of parsing (Charniak, 2000). We argue that this correlation between better approximation and better accuracy suggests that ISBNs are a good abstract model for structure prediction.

2. Inference with Sigmoid Belief Networks

A Sigmoid Belief Network (SBN) (Neal, 1992) is a type of Bayesian Network with binary variables and conditional probability distributions in the form:

$$P(S_i = 1|Par(S_i)) = \sigma(\sum_{S_j \in Par(S_i)} J_{ij} S_j),$$

where $Par(S_i)$ are the parents of S_i, σ denotes the logistic sigmoid function, and J_{ij} is the weight for the arc from variable S_j to variable S_i. SBNs are similar to feed-forward neural networks, but unlike neural networks SBNs have a precise probabilistic semantics of their hidden variables. In this paper we consider a generalized version of SBNs where we allow variables with any range of discrete values. The normalized exponential function is used to define the conditional probability distributions at these nodes.

Exact inference with all but very small SBNs is not tractable. Initially sampling methods were used (Neal, 1992), but this is also not feasible for large networks, especially for the dynamic models of the type described in section 3. Variational methods have also been proposed for approximating SBNs (Saul & Jordan, 1999). The main idea of variational methods (Jordan et al., 1999) is, roughly, to construct a tractable approximate model with a number of free parameters. The free parameters are set so that the resulting approximate model is as close as possible to the original graphical model for a given inference problem.

The simplest example of a variation method is the mean field method, originally introduced in statistical mechanics and later applied to neural networks in (Hinton et al., 1995). Let us denote the set of visible variables in the model by V and hidden variables by $H = h_1,\ldots,h_l$. The mean field method uses a fully factorized distribution $Q(H|V) = \prod_i Q_i(h_i|V)$ as the approximate model, where each Q_i is the distribution of an individual latent variable. The independence between the variables h_i in this approximate distribution Q does not imply independence of the free parameters which define the Q_i. These parameters are set to minimize the Kullback-Leibler divergence between the approximate distribution $Q(H|V)$ and the true distri-

bution $P(H|V)$ or, equivalently, to maximize:

$$L_v = \sum_H Q(H|V) \ln \frac{P(H,V)}{Q(H|V)}. \quad (2)$$

The expression L_v is a lower bound on the log-likelihood $\ln P(V)$. It is used in the mean field theory (Saul & Jordan, 1999) as an approximation of the likelihood. However, in our case of dynamic graphical models, we have to use a different approach which allows us to construct an incremental structure prediction method without needing to introduce the additional parameters proposed in (Saul & Jordan, 1999), as we will discuss in section 5.2.

3. Exploiting Structural Locality

As discussed in the introduction, we want to extend SBNs to allow the model structure to depend on the structure being output. In particular, we want the arcs of the model to reflect the same statistical dependencies which are reflected by locality in the output structure. When these arcs connect latent variables, information can be propagated between latent variables, thereby providing an even larger structural domain of locality than that provided by single arcs. This provides a potentially powerful form of feature induction, which is nonetheless biased toward a notion of locality which is appropriate for the problem.

To extend SBNs for processing arbitrarily long sequences such as the decision sequence $D^1, ..., D^m$, we use dynamic models. This gives us a kind of Dynamic Bayesian Network (DBN). In DBNs, a new set of variables is instantiated for each position in the sequence, but the arcs and weights for these variables are the same as in other positions. The arcs which connect variables instantiated for different positions must be directed forward in the sequence, thereby allowing a temporal interpretation of the sequence.

In order to have arcs which reflect locality in the output structure, we need to specify arcs based on the actual outputs of the decision sequence, not based on adjacency in the sequence. We allow a decision to effect the placement of any arc whose destination is after the decision. This gives us a form of switching model (Murphy, 2002), where each decision switches the model structure used for the remaining decisions. The incoming arcs for a given position are a discrete function of the sequence of decisions which precede that position. For this reason we call our model an "incremental" model, not just a dynamic model. The structure of the model is determined incrementally as the decision sequence proceeds.

Incremental Sigmoid Belief Networks allow the model structure to depend on the output structure without overly complicating the inference of the desired conditional probabilities $P(D^t|D^1, \ldots, D^{t-1})$. At position t in the sequence, the only arcs whose placement are not specified by $D^1, \ldots, D^{t-1}$ have their destinations after t. Also, there are no visible variables after t. Therefore none of the arcs whose placement is not yet known can have any impact on the inference of $P(D^t|D^1, \ldots, D^{t-1})$. This is why in figure 1, discussed below, it is not necessary to try to draw the portion of the graph after t. This property of ISBNs allows us to do inference without the need to sum over all possible model structures, which in general would make inference intractable. Note that this property would not hold if we used an undirected graphical model, such as Conditional Random Fields.

4. The Probabilistic Model of Structure Prediction

In this section we complete the definition of Incremental Sigmoid Belief Networks for structure prediction. We only consider joint probability models, since they are generally simpler and, unlike history-based conditional models, do not suffer from the label bias problem (Bottou, 1991). Also, in many complex predication tasks, such as phrase structure parsing, all the most accurate models make use of a joint model (Charniak & Johnson, 2005; Henderson, 2004).

We use a history-based probability model, as in equation (1), but instead of treating each D^t as an atomic decision, it is convenient to further split it into a sequence of elementary decisions $D^t = d_1^t, \ldots, d_n^t$:

$$P(D^t|D^1, \ldots, D^{t-1}) = \prod_k P(d_k^t|h(t,k)),$$

where $h(t,k)$ denotes the decision history $D^1, \ldots, D^{t-1}, d_1^t, \ldots, d_{k-1}^t$. For example, a decision to create a new node in a labeled output structure can be divided into two elementary decisions: deciding to create a node and deciding which label to assign to it.

An example of the kind of graphical model we propose is illustrated in figure 1. It is organized into vectors of variables: latent state variable vectors $S^{t'} = s_1^{t'}, \ldots, s_n^{t'}$, representing an intermediate state at position t', and decision variable vectors $D^{t'}$, representing a decision at position t', where $t' \leq t$. Variables whose value are given at the current decision (t,k) are shaded in figure 1, latent and current decision variables are left unshaded.

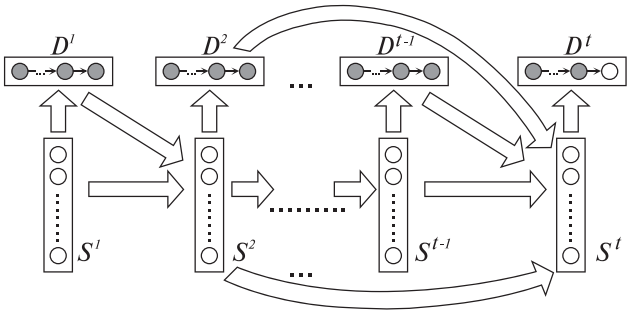

Figure 1. ISBN for estimating $P(d_k^t|h(t,k))$.

As illustrated by the arcs in figure 1, the probability of each state variable $s_i^{t'}$ depends on all the variables in a finite set of relevant previous state and decision vectors, but there are no direct dependencies between the different variables in a single state vector. Which previous state and decision vectors are connected to the current state vector is determined by a set of structural relations specified by the model designer. For example, we could select the most recent state where the same output structure node was on the top of the processor's stack, and a decision variable representing that node's label. Each such selected relation has its own distinct weight matrix for the resulting arcs in the graph, but the same weight matrix is used at each position where the relation is relevant.

As indicated in figure 1, the probability of each elementary decision $d_k^{t'}$ depends both on the current state vector $S^{t'}$ and on the previously chosen elementary action $d_{k-1}^{t'}$ from $D^{t'}$. This probability distribution has the form of a normalized exponential:

$$P(d_k^{t'} = d|S^{t'}, d_{k-1}^{t'}) = \frac{\Phi_{h(t',k)}(d)\, e^{\sum_j W_{dj} s_j^{t'}}}{\sum_{d'} \Phi_{h(t',k)}(d')\, e^{\sum_j W_{d'j} s_j^{t'}}}, \quad (3)$$

where $\Phi_{h(t',k)}$ is the indicator function of the set of elementary decisions that may possibly follow the last decision in the history $h(t',k)$, and the W_{dj} are the weights of the arcs from the state variables.

5. Approximating Inference in ISBNs

Exact inference with ISBNs is straightforward, but not tractable, so we need to develop methods for approximating the inference problems required for structure prediction. Gibbs sampling is also absolutely infeasible because of the huge space of variables and need to resample after making each new decision in the sequence. Thus, we know of no reasonable alternatives to the use of variational methods.

5.1. A Feed-Forward Approximation

In this section we will introduce the application of variational methods to ISBNs, and present the sense in which neural network computation can be regarded as a mean field approximation, under the additional constraint of strictly feed-forward computation. We will call this approximation the feed-forward approximation. As in any mean field approximation, each of the latent variables in the variational model is independently distributed. But unlike the general case of mean field approximation, in the feed-forward approximation we only allow the parameters of the distributions Q_i to depend on the approximate distributions of their parents. This additional constraint increases the potential for a large KL divergence with the true model, but it significantly simplifies the computations.

The set of hidden variables H in our graphical model consists of all the state vectors $S^{t'}$, $t' \leq t$, and the current decision d_k^t. All the previously observed decisions $h(t,k)$ comprise the set of visible variables V. The approximate fully factorisable distribution $Q(H|V)$ can be written as:

$$Q(H|V) = q_k^t(d_k^t) \prod_{t'i} \left(\mu_i^{t'}\right)^{s_i^{t'}} \left(1 - \mu_i^{t'}\right)^{1 - s_i^{t'}}.$$

where $\mu_i^{t'}$ is the free parameter which determines the distribution of state variable i at position t', namely its mean, and $q_k^t(d_k^t)$ is the free parameter which determines the distribution over decisions d_k^t, namely the estimate of $P(d_k^t|h(t,k))$.

Because we are only allowed to use the approximate distributions of the parent variables to compute the free parameters $\mu_i^{t'}$, the optimal assignment is given by $\mu_i^{t'} = \sigma(\eta_i^{t'})$, where $\eta_i^{t'}$ is a weighted sum of the parent variables' means:

$$\eta_i^{t'} = \sum_{t'' \in R(t')} \sum_j J_{ij}^{\tau(t',t'')} \mu_j^{t''} + \sum_k B_{id_k^{t''}}^{\tau(t',t'')}, \quad (4)$$

where $R(t')$ is the set of related previous positions, and $\tau(t',t'')$ is the relevant relation between the position t'' and the position t'.

In order to maximize (2), the approximate distribution of the next decision $q_k^t(d)$ should be set to

$$q_k^t(d) = \frac{\Phi_{h(t,k)}(d)\, e^{\sum_j W_{dj} \mu_j^t}}{\sum_{d'} \Phi_{h(t,k)}(d')\, e^{\sum_j W_{d'j} \mu_j^t}}, \quad (5)$$

as follows from expression (3). The resulting estimate of the structure probability is given by:

$$P(S) \approx \prod_{t,k} q_k^t(d_k^t). \quad (6)$$

This approximation method replicates exactly the computation of the feed-forward neural network in (Henderson, 2003), where the above means $\mu_i^{t'}$ are equivalent to the neural network hidden unit activations. Thus, that neural network probability model can be regarded as a simple approximation to the graphical model introduced in section 4.

In addition to the drawbacks shared by any mean field approximation method, this feed-forward approximation cannot capture top-down reasoning. By top-down reasoning we mean the need to update the state vector means $\mu_i^{t'}$ after observing a decision d_k^t, for $t' \leq t$. The next section discusses how top-down reasoning can be incorporated in the approximate model.

5.2. A Mean Field Approximation

The standard use of the mean field theory for SBNs (Saul & Jordan, 1999) is to approximate probabilities using the value of L_v from expression (2). Unfortunately this is not feasible with ISBNs. To approximate $P(d^t{}_k|h(t,k))$ using the value of L_v, we have to include the current decision d_k^t in the visible variables V, and compute a separate estimate $L_v^{t,k}(d_k^t)$ for each possible value of d_k^t. Then $P(d^t{}_k|h(t,k))$ can be approximated as the normalized exponential of $L_v^{t,k}(d_k^t)$ values. This computation is especially infeasible with labeled output structures, where the number of possible alternative decisions d_k^t can be large, as for example when predicting the words in a phrase structure tree. Even if we choose not to recompute mean field parameters for all the preceding states $S^{t'}$, but only for the current state S^t (as proposed below), tractability still remains a problem.[1]

In our modification of the mean field method, we consider the next decision d_k^t as a hidden variable, as above in the feed-forward approximation. Then the assumption of full factorisability of $Q(H|V)$ is stronger than in the standard mean field theory because the approximate distribution $Q(H|V)$ is no longer conditioned on the next decision d_k^t.

Again as in the feed-forward approximation, we are interested in finding the distribution Q which maximizes the quantity L_v in expression (2). The decision distribution $q_k^t(d_k^t)$ maximizes L_v when it has the same dependence on the state vector means μ_k^t as in the feed-forward approximation, namely expression (5). However, as we mentioned above, the feed-forward computation does not allow us to compute the optimal values of state means $\mu_i^{t'}$.

Optimally, after each new decision d_k^t, we should recompute all the means $\mu_i^{t'}$ for all the state vectors $S^{t'}$, $t' \leq t$. However, this would make the method intractable for tasks with long decision sequences. Instead, after making each decision d_k^t and adding it to the set of visible variables V, we recompute only the means of the current state vector S^t. This approach also speeds up computation because, unlike in the standard mean field theory, there is no need to introduce an additional variational parameter for each hidden layer variable s_i^t.

The denominator of the normalized exponential function in (3) does not allow us to compute L_v exactly. Instead, we approximate the expectation of its logarithm by substituting s_j^t with their means μ_j^t.[2] Unfortunately, even with this assumption there is no analytic way to maximize L_v with respect to the means μ_k^t, so we need to use numerical methods. We can rewrite the expression (2) as follows, substituting the true $P(H,V)$ defined by the graphical model and the approximate distribution $Q(H|V)$, omitting parts independent of μ_k^t:

$$L_v^{t,k} = \sum_i -\mu_i^t \ln \mu_i^t - (1-\mu_i^t) \ln \left(1-\mu_i^t\right) + \mu_i^t \eta_i^t$$

$$+ \sum_{k'<k} \sum_j W_{d^t_{k'} j} \mu_j^t - \ln \left(\sum_d \Phi_{h(t,k')}(d) e^{\sum_j W_{dj} \mu_j^t} \right), \quad (7)$$

here, η_i^t is computed from the previous relevant state means and decisions as in (4). This expression is concave with respect to the parameters μ_i^t, so the global maximum can be found. We use coordinatewise ascent, where each μ_i^t is selected by a line search while keeping other $\mu_{i'}^t$ fixed.

Though we avoided recomputation of means of the previous states, estimation of the complex decision probability $P(D^t|h(t,k))$ will be expensive if the decision D^t is decomposed in a large number of elementary decisions. As an example, consider a situation in natural language dependency parsing, where after deciding to create a link, the parser might need to decide on the type of the link and, then, predict the part of speech type of the word and, finally, predict the word itself. The main reason for this complexity is the presence

[1] We conducted preliminary experiments with natural language parsing on very small datasets and even in this setup the method appeared to be very slow and, surprisingly, not as accurate as the modification considered further in this section.

[2] In initial research, we considered the introduction of additional variational parameters associated with every possible value of the decision variable in a way similar to (Saul & Jordan, 1999), but this did not improve the prediction accuracy of the model, and considerably increased the computational time.

of the summation over k' in expression (7), which results in expensive computations during the search for an optimal value of μ_i^t. This computation can be simplified by using the means of S^t computed during the estimation of $P(d_{k-1}^t|h(t,k-1))$ as priors for the computation of the same means during the estimation of $P(d_k^t|h(t,k))$. If we denote the means computed at an elementary step (t,k) as $\mu_i^{t,k}$, then for $k=1$, minimization of $L_v^{t,k}$ can be performed analytically, by setting $\mu_i^{t,1}$ to $\sigma(\eta_i^{t'})$. For $k>1$, expression (7) can be rewritten as:

$$L_v^{t,k} = \sum_i -\mu_i^{t,k}\ln\mu_i^{t,k} - (1-\mu_i^{t,k})\ln\left(1-\mu_i^{t,k}\right)$$
$$+\mu_i^{t,k}\left(\ln(\mu^{t,k-1}) - \ln(1-\mu^{t,k-1})\right) + \sum_j W_{d_{k-1}^t j}\mu_j^{t,k}$$
$$-\ln\left(\sum_d \Phi_{h(t,k-1)}(d)\exp(\sum_j W_{dj}\mu_j^{t,k})\right). \quad (8)$$

After computing the last decision k for the state t, means μ^t are computed in a similar way. These means μ^t are then used in the computation of $\eta_i^{t'}$ (4) for the relevant future states t', $t \in R(t')$.

5.3. Learning

We train the models described in sections 5.1 and 5.2 to maximize the fit of the *approximate* models to the data. We use gradient descent, and a maximum likelihood objective function. In order to compute the derivatives with respect to the model parameters, the error should be propagated back through the structure of the graphical model. For the feed-forward approximation, computation of the derivatives is straightforward, as in neural networks. But for the mean field approximation, this requires computation of the derivatives of the means μ_i^t with respect to the other parameters in expressions (7) and (8). The use of a numerical search in the mean field approximation makes the analytical computation of these derivatives impossible, so a different method needs to be used to compute their values. If minimization of $L_v^{t,k}$ is done until convergence, then the derivatives of $L_v^{t,k}$ with respect to μ_i^t are close to zero. This gives us a system of linear equations, which describes interdependencies between the current means, the means of the related previous states, and the weights. Then, implicit differentiation can be used to compute the needed derivatives.

The standard mean field approach considered in (Saul & Jordan, 1999) maximized L_v during learning, because L_v was used as an approximation of the log-likelihood of the training data. L_v is actually the sum of the log-likelihood and the negated KL divergence between the approximate distribution $Q(H|V)$ and the SBN distribution $P(H|V)$. Thus, maximizing L_v will at the same time direct the SBN distribution toward configurations which have a lower approximation error. It is important to distinguish this regularization of the approximate distribution from the usual regularization of the SBN distribution, which can be achieved by simple weight decay. We believe that these two regularizations should be complimentary. However, in our version of the mean field method the approximate distributions of hidden decision variables q_k^t are used to compute the data likelihood (6) and, thus, maximizing this target function will not automatically imply KL divergence minimization. Application of an additional regularization term corresponding to minimization of the KL divergence might be beneficial for our approach, and it could be a subject of further research. In our current experiments, we used standard weight decay, which regularizes the SBN distribution with a Gaussian prior over weights.

6. Experiments

The goal of the evaluation is to demonstrate that incremental SBNs are an appropriate model for structure prediction. Also, we would like to show that learning the mean field approximation derived in section 5.2 (MF method) results in a sufficiently accurate model, and that this model is more accurate than the feed-forward neural network approximation (NN method) of (Henderson, 2003) considered in section 5.1. First, we start with an artificial experiment where the true distribution is generated by an SBN, and compare both of the approximation models *learned* on this artificial data. Second, we apply the models to a real problem, parsing of natural language, where we compare our approximations with state-of-the-art models.

6.1. Artificial Experiment

In order to have an upper bound for our artificial experiments, we do not consider incremental models but use a dynamic Sigmoid Belief Network, a first order Markov model, and consider a sequence labeling task. This simplification allowed us to use Gibbs sampling from a *true* model as an upper bound of accuracy. We generated the training data from random dynamic SBNs of the following type: first a label Y^t is sampled from the distribution $P(Y^t|S^t)$ as in (3), then an input element X^t is sampled from the distribution $P(X^t|Y^t,S^t)$. Different weight matrices were used in the computation of $P(X^t|Y^t,S^t)$ for each value of the label Y^t. The state size was set to 5, the number of possible labels to 6, and the number of distinct in-

Table 1. Percentage labeled constituent recall (R), precision (P), combination of both (F_1) on the testing set.

	R	P	F_1
Bikel, 2004	87.9	88.8	88.3
Taskar et al., 2004	89.1	89.1	89.1
NN method	**89.1**	**89.2**	**89.1**
Turian and Melamed, 2006	89.3	89.6	89.4
MF method	**89.3**	**90.7**	**90.0**
Charniak, 2000	90.0	90.2	90.1

put elements to 8. We performed 10 experiments.[3] For each of the experiments, we trained both MF and NN approximations on training sequence of 20,000 elements, and tested them on another 10,000 elements. Weight-decay and learning rate were reduced through the course of the experiments whenever accuracy on the development set went down. Beam search with a beam of 10 was used during testing. The MF methods achieved average error reduction of 27% with respect to the NN method, where accuracy of the Gibbs sampler was used as an upper bound (average accuracies of 80.5%, 81.0%, and 82.3% for the NN, MF, and sampler, respectively).

The MF approximation performed better than the NN approximation on 9 experiments out of 10 (statistically significant in 8 cases). These results suggest that the MF method leads to a much more accurate model when the true distribution is defined by a dynamic SBN. In addition, the average relative error reduction of even the NN approximation over the unigram model exceeded 60% (the unigram model accuracy was 77.4% on average), which suggests that both approximations are sufficiently accurate and learnable.

6.2. Natural Language Parsing

We compare our two approaches on a natural language problem, the phrase structure parsing task. The output structure is defined as a labeled tree, which specifies the hierarchical decomposition of a sentence into phrases. The hypothesis we wish to test here is that the more accurate approximation of ISBNs will result in a more accurate model of parsing. If this is true, then it suggests that ISBNs are a good abstract model for structure prediction, or at least for problems similar to natural language parsing.

We used the Penn Treebank Wall Street Journal corpus to perform the empirical evaluation of the considered approaches. It is expensive to train the MF approximation on the whole WSJ corpus, so instead we used only sentences of length at most 15, as in (Taskar et al., 2004) and (Turian et al., 2006). The standard split of the corpus into training (9,753 sentences, 104,187 words), validation (321 sentences, 3,381 words), and testing (603 sentences, 6,145 words) was performed. We replicated the same definition of derivation and the same pattern of interconnection between states as described in (Henderson, 2003).

During parsing with both the NN method and the MF method, we used beam search with a post-word beam of 10. Increasing the beam size beyond this value did not significantly effect parsing accuracy. For both of the models, the state vector size of 40 was used. All the parameters for both the NN and MF models were tuned on the validation set. A single best model of each type was then applied to the final testing set.

Table 1 lists the results of the NN approximation and the MF approximation,[4] along with results of different generative and discriminative parsing methods evaluated in the same experimental setup (Turian et al., 2006; Charniak, 2000). The MF model improves over the baseline NN approximation, with a relative error reduction in F-measure exceeding 8%. This improvement is statically significant. The MF model achieves results which do not appear to be significantly different from the results of the best model in the list (Charniak, 2000). It should also be noted that the model of (Charniak, 2000) is the most accurate history-based probabilistic model on the standard WSJ parsing benchmark, which confirms the viability of our model.

These experimental results suggest that ISBNs are an appropriate model for structure prediction. Even approximations such as those tested here, with a very strong factorisability assumption, allow us to build quite accurate parsing models.[5] We believe this provides strong justification for work on more accurate approximations of ISBNs.

7. Related Work

Whereas graphical models are standard models for sequence processing, there has not been much previ-

[3] We preselected these 10 models to avoid random dynamic SBNs with trivial distributions. We excluded SBNs for which unigram model accuracy was within 3% of the Gibbs sampler accuracy, and where accuracy of the Gibbs sampler did not exceed 70%. All these constants were selected before conducting the experiments.

[4] Approximate training times on a standard desktop PC for the MF and NN approximations were 140 and 3 hours, respectively, and parsing times were 3 and 0.05 seconds per token, respectively. Parsing with the MF method could be made more efficient, for example by not requiring the numerical approximations to reach convergence.

[5] We plan to make our implementation of the parser publicly available soon.

ous work on graphical models for prediction of structures more complex than sequences. Latent variable models, including undirected graphical models, were successfully applied to the task of structure reranking, e.g. (Koo & Collins, 2005). Dependency parsing with Dynamic Bayesian Networks was considered in (Peshkin & Savova, 2005), with limited success. Roughly, the model considered the whole sentence at a time, with the DBN being used to decide which words correspond to leaves of the tree. The chosen words are then removed from the sentence and the model is recursively applied to the reduced sentence.

Sigmoid Belief Networks were used originally for character recognition tasks, but later a Markovian dynamic extension of this model was applied to the reinforcement learning task (Sallans, 2002). However, their graphical model, approximation method, and learning method differ substantially from those of this paper.

8. Conclusions

This paper proposes a new class of model for structure prediction problems, Incremental Sigmoid Belief Networks. These graphical models allow the structure of the model to be dependent on the output structure, which allows the induction of latent variables with a structural locality bias appropriate for the domain. Exact inference with the proposed class of graphical models is not tractable, but we derive two tractable approximations. First, it is shown that the feed-forward neural network of (Henderson, 2003) can be considered as a simple approximation to ISBNs. Second, a more accurate but still tractable approximation based on mean field theory is proposed.

Both approximation models are empirically evaluated. First, artificial experiments were performed, where both approximations significantly outperformed a baseline. The mean field method achieved average relative error reduction of about 27% over the neural network approximation, demonstrating that it is a more accurate approximation. Second, both approximations are applied to the natural language parsing task, where the mean field method demonstrated significantly better results. These results are non-significantly different from the results of the most accurate history-based probabilistic model of parsing (Charniak, 2000). The fact that a more accurate approximation leads to a more accurate parser suggests that the ISBNs proposed here are a good abstract model for structure prediction. This empirical result motivates further research into more accurate approximations of ISBNs.

References

Bottou, L. (1991). *Une approche théoretique de l'apprentissage connexionniste.* Doctoral dissertation, Univ. Paris XI, Paris, France.

Charniak, E. (2000). A maximum-entropy-inspired parser. *Proc. North American Chap. Assoc. Comp. Linguistics.* Seattle, Washington.

Charniak, E., & Johnson, M. (2005). Coarse-to-fine n-best parsing and MaxEnt discriminative reranking. *Proc. Meeting of Assoc. Comp. Linguistics.* Ann Arbor, MI.

Collins, M. (1999). *Head-driven statistical models for natural language parsing.* Doctoral dissertation, Univ. Pennsylvania, Philadelphia, PA.

Durbin, R., Eddy, S. R., Krogh, A., & Mitchison, G. (2003). *Biological sequence analysis: Probabilistic models of proteins and nucleic acids.* Cambridge, UK.

Henderson, J. (2003). Inducing history representations for broad coverage statistical parsing. *Proc. North American Assoc. Comp. Linguistics.* Edmonton, Canada.

Henderson, J. (2004). Discriminative training of a neural network statistical parser. *Proc. 42nd Meeting of Assoc. Comp. Linguistics.* Barcelona, Spain.

Hinton, G., Dayan, P., Frey, B., & Neal, R. (1995). The wake-sleep algorithm for unsupervised neural networks. *Science*, *268*.

Jordan, M. I., Z.Ghahramani, Jaakkola, T. S., & Saul., L. K. (1999). An introduction to variational methods for graphical models. In M. I. Jordan (Ed.), *Learning in graphical models.* Cambridge, MA: MIT Press.

Koo, T., & Collins, M. (2005). Hidden-variable models for discriminative reranking. *Proc. Conf. on Empirical Methods in NLP.* Vancouver, B.C., Canada.

Murphy, K. P. (2002). *Dynamic Bayesian networks: Representation, inference and learning.* Doctoral dissertation, Univ. California, Berkeley, CA.

Neal, R. (1992). Connectionist learning of belief networks. *Artificial Intelligence*, *56*, 71–113.

Peshkin, L., & Savova, V. (2005). Dependency parsing with dynamic Bayesian network. *AAAI.* Pittsburgh, PA.

Sallans, B. (2002). *Reinforcement learning for factored markov decision processes.* Doctoral dissertation, Univ. Toronto, Toronto, Canada.

Saul, L. K., & Jordan, M. I. (1999). A mean field learning algorithm for unsupervised neural networks. In M. I. Jordan (Ed.), *Learning in graphical models*, 541–554. Cambridge, MA: MIT Press.

Taskar, B., Klein, D., Collins, M., Koller, D., & Manning, C. (2004). Max-margin parsing. *Proc. Conf. on Empirical Methods in NLP.* Barcelona, Spain.

Turian, J., Wellington, B., & Melamed, D. (2006). Scalable discriminative learning for natural language parsing and translation. *Proc. NIPS.* Vancouver, Canada.