Learning in Markov Random Fields with Contrastive Free Energies

Max Welling
School of Information and Computer Science
University of California Irvine
Irvine CA 92697-3425 USA
welling@ics.uci.edu

Charles Sutton
Department of Computer Science
University of Massachusetts
Amherst, MA 01002
casutton@cs.umass.edu

Abstract

Learning Markov random field (MRF) models is notoriously hard due to the presence of a global normalization factor. In this paper we present a new framework for learning MRF models based on the *contrastive free energy* (CF) objective function. In this scheme the parameters are updated in an attempt to match the average statistics of the data distribution and a distribution which is (partially or approximately) "relaxed" to the equilibrium distribution. We show that maximum likelihood, mean field, contrastive divergence and pseudo-likelihood objectives can be understood in this paradigm. Moreover, we propose and study a new learning algorithm: the "k-step Kikuchi/Bethe approximation". This algorithm is then tested on a conditional random field model with "skip-chain" edges to model long range interactions in text data. It is demonstrated that with no loss in accuracy, the training time is brought down on average from 19 hours (BP based learning) to 83 minutes, an order of magnitude improvement.

1 INTRODUCTION: LEARNING MRFs

In the context of machine learning two classes of graphical model have been extensively studied: the directed graphical model or Bayesian network (BN) and the undirected graphical model or Markov random field (MRF). While both models have been applied successfully in a number of domains, it is fair to say that learning in BNs has reached a more advanced level of sophistication than learning in MRFs. For instance, hidden variable models can be efficiently tackled with the variational EM algorithm[1], Bayesian inference is often feasible with conjugate priors and greedy structure learning algorithms have met with some success as well. In contrast, even for a fully observed MRF model, evaluating the gradient of the log-likelihood is typically intractable. The problem can be traced back to the presence of a *global* normalization term which depends on the parameters and which translates into an often intractable inference problem when we compute its gradient[2]. Clearly, introducing unobserved random variables only aggravates this problem, while Bayesian approaches to infer posterior distributions over parameters or structures seem completely absent in the literature, apart from one paper [9]. Because MRF models arise in many applications, including spatial statistics, computer vision, and natural-language processing, we feel that it is important to improve this state of affairs.

We claim that learning MRFs is so difficult because the inference problem induced by the global normalizer is of a different nature and often harder than the problem of computing the posterior distribution of the hidden variables *given* the observed variables needed for learning BNs. The reason is that in the latter case we enter evidence to the model and we may have reasonable hope that the posterior is peaked around a single solution. However, for MRFs we need to infer the distribution when all variables are unconstrained implying that the distribution we are trying to infer is likely to have many modes. Even though much progress has been made in the field of approximate inference, no method can satisfactorily deal with a large number of modes for which the location is unknown.

To approximate the required averages over the unconstrained (model) distribution we could for instance run a MCMC sampler or use the mean field approximation [10]. While the first method is relatively slow (we need to sample for every iteration of gradient descent), the estimated statistics can also get swamped by the sampling variance[3].

[1]Fully observed BNs are trivial and only depend on counts.

[2]In case the structure of the graph is such we can identify a junction tree with small tree-width, then inference can be performed tractably and we can compute exact learning rules.

[3]Of course, one can reduce the variance by using more samples, but note that this only improves as $1/N$ where N is the number of samples.

The mean field approximation is not plagued by variance, but unlike the MCMC sampler it has to tolerate a certain bias in its estimates. However, both problems suffer from a much more severe problem, namely that they will only approximate one mode of the distribution. One could argue that a "good sampler" should mix between modes, but in the absence of any information about the location of these modes, this is an unrealistic hope, certainly in high dimensions.

There is one piece of information which typically remains unexploited, namely the fact that data points are expected to be located close to a mode (or at least this is what we like to achieve during learning). Hence, one idea to deal with the above mentioned "many modes" problem, is to run multiple MCMC chains, each one initialized at a different data-point. With this method, we are at least certain that all the modes close to data points are explored by samples. This will have the effect that learning is likely to get the local shape of each local mode correct. Still, there are (at least) two drawbacks: 1) the modes do not communicate, i.e. we have no mixing between modes and 2) accidental modes which are created because of the particular parameterization of the model remain undetected by samples implying there is no force to remove them from the model. The first problem has the undesirable effect that although the shape of each mode may be a good fit, the relative volume (or free energy) of the modes may not be properly estimated. This was studied in [7] and mode-jumping MCMC procedures were proposed to improve the communication between modes. Since there is no information about the location of the spurious modes (mentioned under 2), we predict it will be extremely hard to deal with the second problem.

Running Markov chains to convergence at every data case at every iteration of learning is clearly a costly business. Fortunately, it turns out that we can greatly improve our efficiency by running these Markov chains for only a few (say k) steps[4]. It turns out that if one uses these pseudo-samples, or rather "k-step reconstructions" of the data, we approximately minimize the so-called "contrastive divergence" objective function [5]. Apart from a very significant increase in efficiency, we also decrease the variance of our estimates at the expense of an increased bias.

The aim of the current paper is to combine deterministic, variational approximations with the ideas of contrastive divergence. This idea is analogous to the introduction of mean field learning in MRFs in [10]. A mean field based approach to contrastive divergence was presented in [17]. In the current work we extend these ideas to general variational approximations. In particular we study the Bethe approximation, which in combination with the convergent "belief optimization" algorithm to minimize the Bethe free energy results in a novel algorithm to train Markov random fields with loopy structure. This algorithm is tested on a conditional random field model with long range interactions (the so called "skip-chain" CRFs [11]) to label tokens in email messages. We demonstrate that we can speed up learning tenfold at no cost to the test-performance of the trained model.

[4] It is essential that the chains are started at the data-cases.

2 MAXIMUM LIKELIHOOD LEARNING

An intuitive way to restate the maximum likelihood objective is as a minimization problem of the following Kullback-Leibler divergence between the data distribution $P_0(y)$ and the model distribution $P_\lambda(y)$,

$$\lambda^{ML} = \arg\min_\lambda KL\left[P_0(y)||P_\lambda(y)\right] \tag{1}$$

We will consider the general case here, where apart from the observed variables, y, the model may also contain a number of unobserved variables h. Introducing the joint distribution $P_\lambda(y,h)$ and the distribution $P_0(y,h) = P_\lambda(h|y)P_0(y) = P_\lambda(h,y)P_0(y)/P_\lambda(y)$ with $P_\lambda(y) = \sum_h P_\lambda(y,h)$, we can rewrite the KL divergence as a difference between two free energies,

$$\begin{aligned} KL[P_0(y)||P_\lambda(y)] &= KL[P_0(y,h)||P_\lambda(y,h)] \\ &= F_0 - F_\infty \doteq \mathbf{CF}_\infty \geq 0 \end{aligned} \tag{2}$$

where F_0 denotes the free energy of the distribution $P_0(y,h)$, while $F_\infty = -\log(Z)$ denotes the free energy of the "random system" governed by P_λ. The subscript ∞ indicates that we have to run a Markov chain infinitely long to reach equilibrium. For every data-case we can therefore identify two random systems; one system with free energy F_0 has a data case clamped to the observed random variables while the hidden variables are free to fluctuate. In the "free system" (with free energy F_∞) all random variables (y,h) are unconstrained. The energy of the system, $E(y,h)$, is defined through the Boltzman distribution,

$$P(y,h) = \frac{1}{Z}\exp\left[-E(y,h)\right]. \tag{3}$$

Although our discussion is more general, we will restrict ourselves from now on to exponential family distributions defined through the following energy function,

$$E(y,h) = -\sum_\beta \sum_i \lambda_{i\beta} f_{i\beta}(y_\beta, h_\beta). \tag{4}$$

In analogy to physical systems, we can decompose the free energy in an average energy term and a entropy term,

$$F_0 = \mathbb{E}[E]_0 - H_0 \qquad F_\infty = \mathbb{E}[E]_\infty - H_\infty \tag{5}$$

where $\mathbb{E}[\cdot]_0$ denotes averaging with respect to the joint $P_0(y,h)$ and $\mathbb{E}[\cdot]_\infty$ denotes averaging with respect to the equilibrium distribution $P_\lambda(\mathbf{v},\mathbf{h})$.

Learning can now be understood as follows: for each data case we first compute the free energy F_0 of the system with the datum clamped to the observed units (this involves inference over the hidden units). Then we set the constraints on the observed variables free and let the system relax into a new distribution $P_\lambda(y,h)$ with lower free energy F_∞. If in this process the expected sufficient statistics $\mathbb{E}[f_{i\beta}]$ change we have an imperfect model and we change the parameters $\lambda_{i\beta}$ in such a way that the expected sufficient statistics are better preserved in the next iteration,

$$\frac{\partial \mathbf{CF}_\infty}{\partial \lambda_{i\beta}} = -\mathbb{E}[f_{i\beta}(y_\beta,h_\beta)]_{P_0} + \mathbb{E}[f_{i\beta}(y_\beta,h_\beta)]_{P_\lambda} \quad (6)$$

Note that this does not mean that the statistics for each data point must cancel with the equilibrium statistics; this property must only hold when averaged over all data cases.

3 APPROXIMATE ML-LEARNING

In the previous section, we wrote the likelihood function as a difference of two free energies, one of which was intractable to compute in general. In this section, we replace those free energies with approximate free energies, in a way conceptually similar to the mean field approximation introduced in [10]. The idea is to replace the objective in Eqn.2 with

$$\begin{aligned} &KL[Q_0(y,h)||P_\lambda(y,h)] - KL[Q_\infty(y,h)||P_\lambda(y,h)] \\ &= F_0^{\text{APP}} - F_\infty^{\text{APP}} \doteq \mathbf{CF}_\infty^{\text{APP}} \geq 0 \end{aligned} \quad (7)$$

where we define $Q_0(y,h) = Q(h|y)P_0(y)$ and where both $Q_0(h|y)$ and $Q(y,h)$ are approximate, variational distributions such as fully factorized mean field distributions or tree structured distributions. Typically they depend on a number of variational parameters that need to be computed by separately minimizing the respective KL-divergence terms in Eqn.7. The most important simplification that is achieved by minimizing $\mathbf{CF}^{\text{APP}}$ is the fact that the log-partition function term, $\log Z$, cancels between the two terms in Eqn.7.

An important constraint that must be satisfied by any contrastive free energy is that $F_0 \geq F_\infty$ or equivalently $\mathbf{CF} \geq 0$. The reason is that we like to change the unconstrained system with F_∞ so that on average it is similar to the constrained system with F_0. This would ensure that if we sample from P_λ the samples would be similar to the data-cases. Since both systems have the same energy function, but an unconstrained system has more entropy its free energy should be lower as well (see Eqn.5). Moreover, the cost function $F_0 - F_\infty$ wouldn't be lower bounded if F_∞ was allowed to become arbitrarily large.

As an example, let's choose the mean field approximation for $Q_0(h|y)$ and $Q_\infty(h,y)$ in Eqn.7 above,

$$Q_0(h|y) = \prod_i q_i(h_i|y) \qquad Q_\infty(y,h) = \prod_j r_j(z_j) \quad (8)$$

with $z = \{y,h\}$ and where both q and r are variational parameters satisfying $\sum_{h_i} q(h_i|y) = 1\ \forall i$ and $\sum_{z_j} r(z_j) = 1\ \forall j$. They are computed by minimizing their respective KL-divergence terms in Eqn.7. It is now easy to see that F_∞ is smaller than F_0, simply because it has more degrees of freedom to minimize over (in F_0 the variables y are constrained). It is convenient to imagine a process where we minimize F_∞ in two phases, first we clamp y to a data-case and minimize over h, then we set the y variables free and continue the minimization over (y,h) jointly[5]. Once we have found the variational parameters (q,r), we can update the parameters using the following gradient,

$$\frac{\partial \mathbf{CF}_\infty^{\text{APP}}}{\partial \lambda_{i\beta}} = -\mathbb{E}[f_{i\beta}(y_\beta,h_\beta)]_{Q_0} + \mathbb{E}[f_{i\beta}(y_\beta,h_\beta)]_{Q_\infty} \quad (9)$$

We only need to have access to (approximate) marginal distributions $p_\beta(y_\beta,h_\beta)$ in order to compute the expectations in Eqn.9. Hence, we are allowed to consider general approximate free energies F_0, F_∞ as functions of local marginal distributions only, as long as we can assert that $F_0 \geq F_\infty$. An important example of this is the family of Kikuchi free energies $F^{\text{KIK}}(\{q_\alpha\})$, where the approximate marginals need not be consistent with a global distribution Q. In other words, there may not exist a global distribution Q such that its marginals over clusters of nodes are given by the q_α which minimize F^{KIK}.

The contrastive Kikuchi free energy can be expressed as a sum over constrained local KL-divergences as follows,

$$\begin{aligned} &\mathbf{CF}_\infty^{\text{KIK}} \doteq F_0^{\text{KIK}} - F_\infty^{\text{KIK}} = \\ &\sum_\alpha c_\alpha KL[p_0(y_\alpha)q_\alpha(h_\alpha|y_\alpha)||p_\alpha(y_\alpha,h_\alpha)] - \\ &\sum_\alpha c_\alpha KL[r_\alpha(y_\alpha,h_\alpha)||p_\alpha(y_\alpha,h_\alpha)] \end{aligned} \quad (10)$$

where $p_\alpha(z_\alpha) = \frac{1}{Z_\alpha}\prod_{\beta\subset\alpha}\Psi_\beta(x_\beta)$, and where the set of clusters $\{\alpha\}$ consists of a number of overlapping large clusters which cover the graph such that any interaction Ψ_β fits in one of these clusters. By $p_0(y_\alpha)$ we mean the marginal data distribution over the variables[6] y in cluster α. Since this distribution is fixed, we only minimize over the q_α variables in the first term. The counting numbers c_α make sure that every variable and interaction is effectively counted once. Unlike the mean field approximation, the marginals are overlapping and are required to satisfy certain "marginalization constraints" on the intersections,

$$\sum_{z_\alpha \backslash z_\beta} r_\alpha(z_\alpha) = r_\beta(z_\beta) \quad (11)$$

and similarly for q. We refer to [19] for more details.

[5] In fact, the mean field equations, when run sequentially (one variable at a time), are a form of coordinate descent.

[6] Note that if we write (y_α, h_α) we mean all the variables y and h which reside in cluster α.

In the following we will be working with clusters consisting of edges and nodes only, called the "Bethe approximation", but we like to emphasize that the formalism is easily adapted to general Kikuchi approximations, or in fact region graph approximations [19]. The counting numbers in this case are given by,

$$c_{\text{edge}} = 1, \qquad c_{\text{node}} = 1 - \#\text{neighbors} \tag{12}$$

The approximate learning procedure is again similar to what we have seen before: first we compute the variational parameters (q_α, r_α) by minimizing the respective KL-divergence terms, and subsequently we update the parameters using the following gradients,

$$\frac{\partial \mathbf{CF}_\infty^{\text{BETHE}}}{\partial \lambda_{i\beta}} = -\mathbb{E}[f_{i\beta}(y_\beta, h_\beta)]_{q_\alpha p_0} + \mathbb{E}[f_{i\beta}(y_\beta, h_\beta)]_{r_\alpha} \tag{13}$$

where we need that $\beta \subseteq \alpha$.

When the free energies F_0^{BETHE} and F_∞^{BETHE} are convex in the variational parameters (q, r), we can use a class of algorithms under the name (generalized) belief propagation to minimize the Bethe free energies (or KL-divergences) in Eqn.10. However, the Bethe free energy is only convex under very special circumstances, e.g. when the graph has at most one loop. In general it is littered with local minima and for reasons explained before it does not deserve recommendation to run BP and end up in some random local minimum. Instead, we would like to initialize our minimization procedures on the data-cases. However, it is not clear how to efficiently find a set of messages that will produce a prescribed set of marginals, implying that we have little control over our initialization. Fortunately, algorithms have been developed that do not rely on messages but directly minimize the Bethe free energy as a function of the marginals [14, 20, 4]. In general, these algorithms iteratively construct a convex upper bound on the Bethe free energy and minimize those under the constraints of marginal consistency. Unfortunately, every constrained bound optimization step is a slow iterative algorithm with linear converge in general. Hence, if we use such an algorithm at every step of learning for every data-case we end up with a computationally very inefficient learning algorithm. For binary random variables with pairwise interactions the situation is considerably better, since it was shown in [18] that the constraints can be solved analytically, leaving only the node marginals as free variational parameters. Hence, a truly efficient learning procedure is currently only available for this case, but we are confident that efficient minimization algorithms for the more general case will be developed in the near future.

4 APPROXIMATE CONTRASTIVE FREE ENERGIES

We will now introduce a second approximation that is based on the ideas behind contrastive divergence and combine them with the variational approximations described in the previous section. This will have the effect of making the learning algorithm computationally much more efficient.

Recall our interpretation of learning using a contrastive free energy. First we compute the free energy F_0 at the data-case under consideration and compute the necessary sufficient statistics. Then we relax the constraints on the variables which were clamped to the value of the data-case and let the system reach equilibrium where we compute the values of the sufficient statistics again. The system is relaxed by "hitting" the data distribution P_0 with a transition kernel that has P_λ as its invariant distribution,

$$\begin{aligned} P_1(h, y) &= \sum_{h', y'} \mathcal{K}(h, y | h', y') P_0(h', y') && (14) \\ P_\lambda &= (\mathcal{K})^\infty P_0 && (15) \end{aligned}$$

In practice we replace P_0 by the empirical distribution and achieve the relaxation by running MCMC sampling procedures initialized at the data cases.

The underlying idea of contrastive divergence is that we don't actually have to wait until the system has reached equilibrium, since there is much valuable information in the first few steps of this relaxation process (i.e. after a few steps of the MCMC samplers). If the population of samples have a systematic tendency to move away from the data, we can immediately correct this tendency by changing the parameters such that the probability becomes larger at the location of the data and the probability becomes smaller at the location of the samples obtained after a *brief* MCMC run,

$$\frac{\partial \mathbf{CF}_k^{\text{CD}}}{\partial \lambda_{i\beta}} \approx -\mathbb{E}[f_{i\beta}(y_\beta, h_\beta)]_{P_0} + \mathbb{E}[f_{i\beta}(y_\beta, h_\beta)]_{P_k} \tag{16}$$

Following these gradients downhill approximately minimizes the following contrastive divergence objective,

$$\begin{aligned} & KL[P_0(y, h) || P_\lambda(y, h)] - KL[P_k(y, h) || P_\lambda(y, h)] \\ & = F_0 - F_k \doteq \mathbf{CF}_k \geq 0 && (17) \end{aligned}$$

The derivative of this objective w.r.t. $\lambda_{i\beta}$ contains a term $\partial F_k / \partial \lambda_{i\beta}$ in addition to the terms in Eqn.16. However, it is usually small and rarely in conflict with the other terms in the gradient and as result it can be safely ignored [5].

Clearly, learning with contrastive divergence results in a vast improvement in efficiency. Moreover, because for each data-case there is a nearby sample we reduce the variance in the estimates of the sufficient statistic in Eqn.16 (compared to a MCMC sampler at equilibrium) but at the same time we may have introduced bias in our estimates. However, it is not hard to show that for an infinite number of data-cases and a model that is flexible enough to contain the true model, it must be true that there is a fixed point at the correct parameter value, i.e. the first and second term in Eqn.20 will cancel. We refer to [5, 15, 21] for further details on contrastive divergence learning.

It is now a small step to argue for a procedure that combines the variational approximation of the previous section with the ideas of contrastive divergence. Instead of relaxing the free energy using sampling we will relax it by applying a minimization procedure over the variational distributions Q initialized at Q_0 or over the marginals $r_\alpha(z_\alpha)$, initialized at $p_0(y_\alpha)q_\alpha(h_\alpha|y_\alpha)$. Thus, we define the approximate "k-step" contrastive free energy as,

$$\begin{aligned} &KL[Q_0(y,h)||P_\lambda(y,h)] - KL[Q_k(y,h)||P_\lambda(y,h)] \\ &= F_0^{\text{APP}} - F_k^{\text{APP}} \doteq \mathbf{CF}_k^{\text{APP}} \geq 0 \end{aligned} \tag{18}$$

where F_k^{APP} is a function of the variational distribution Q_k. Alternatively, in case of the Kikuchi approximation, we use Eqn.10 and replace the local marginals $r_\alpha(z_\alpha)$ with their k-step counterparts obtained after k steps of minimization on the Kikuchi free energy. Because of its definition the "k-step" contrastive free energy must be positive which, as discussed earlier, is an important constraint for the procedure to work. Taking derivatives w.r.t. to the parameters $\{\lambda_{i\beta}\}$ we find,

$$\frac{\partial \mathbf{CF}_k^{\text{APP}}}{\partial \lambda_{i\beta}} = \frac{\partial F_0^{\text{APP}}}{\partial \lambda_{i\beta}} - \frac{\partial F_k^{\text{APP}}}{\partial \lambda_{i\beta}} - \frac{\partial F_k^{\text{APP}}}{\partial Q_k}\frac{\partial Q_k}{\partial \lambda_{i\beta}} \tag{19}$$

where the last term appears because we didn't minimize the free energy and hence $\partial F_k/\partial Q_k \neq 0$ (unlike $\partial F_0/\partial Q_0 = 0$ and $\partial F_\infty/\partial Q_\infty = 0$). This term is difficult to compute, since we don't have explicit expressions for Q_k in terms of λ_i. Again, it is small and rarely in conflict with the other terms in the gradient so it can be safely ignored (see [17] for experimental evidence of this fact in the case of MF). Hence, ignoring the last term and simplifying the other terms we arrive at the gradient,

$$\frac{\partial \mathbf{CF}_k^{\text{APP}}}{\partial \lambda_{i\beta}} \approx -\mathbb{E}[f_{i\beta}(y_\beta,h_\beta)]_{Q_0} + \mathbb{E}[f_{i\beta}(y_\beta,h_\beta)]_{Q_k} \tag{20}$$

Of course, when we use the Kikuchi approximation we replace the global distributions Q_0 and Q_k in Eqn.20 by local marginals $q_\alpha p_0$ and $r_{\alpha,k}$ as in Eqn.13.

5 RELATION TO PSEUDO-LIKELIHOOD

We have seen that learning in MRFs can be interpreted as minimizing the difference between two free energies, one with the data clamped on the observed variables, the other one with all the variables unconstrained. Importantly, the latter free energy must always be lower than the first one. The various methods differed in the way we allowed the relaxation of the free energy to take place. We have introduced approximate relaxations using variational distributions and partial relaxations where we don't relax all the way to equilibrium. We will now see that the pseudo-likelihood estimator can also be interpreted in this framework (see also [6]).

In [1], the pseudo-likelihood (PL) was introduced to learn MRF models tractably. For a fully observed[7] MRF the PL is given by,

$$\text{PL} = \prod_{n=1}^{N}\prod_{k=1}^{K} p(\hat{y}_{k,n}|\hat{y}_{-k,n}) \tag{21}$$

where y_{-k} denotes all variables except variable y_k, K is the number of variables and N the number of data-cases. This objective is far more tractable than the ML criterion because it only depends on *one dimensional* normalization constants $Z_{k|-k}$. Moreover it was shown that asymptotically this estimator is consistent [3] (but less efficient in the statistical sense than the MLE). We can rewrite minus the log of this objective as a difference of two free energies,

$$\begin{aligned} &\frac{1}{K} KL[P_0||\prod_k P_{k|-k}] = \mathbb{E}[\sum_{i\beta} \lambda_{i\beta} f_{i\beta}(y_\beta)]_{P_0} + \\ &+\frac{1}{K}\sum_k \log Z_{k|-k} = F_0^{\text{PL}} - F_\infty^{\text{PL}} = \mathbf{CF}^{\text{PL}} \geq 0 \end{aligned} \tag{22}$$

where we identify the first term as the average energy and the second as the average one dimensional conditional partition functions. Since the data have no entropy, the first term is the free energy of the data F_0. The second term can be interpreted as a partially unconstrained free energy, where only one variable is relaxed at a time, conditioned on all the others and where the final result is averaged. Hence, like our partial relaxations, the PL-relaxation stays close to the data distribution since at all times we condition on all but one of the variables. The relaxed distribution for one data-case is given by the following mixture,

$$P_\lambda^{\text{PL}}(y_n) = \frac{1}{K}\sum_{k=1}^{K}\left[p_\lambda(y_{k,n}|\hat{y}_{-k,n})\prod_{j\backslash k}\delta(y_{j,n}-\hat{y}_{j,n})\right] \tag{23}$$

which has to be compared with P_λ (maximum likelihood), P_k (k-step contrastive divergence), Q_∞ (variational) and Q_k (k-step variational). It is now straightforward to derive the following gradients,

$$\begin{aligned} \frac{\partial \mathbf{CF}^{\text{PL}}}{\partial \lambda_{i\beta}} &= -\mathbb{E}[f_{i\beta}(y_\beta)]_{P_0} + \mathbb{E}[f_{i\beta}(y_\beta)]_{P^{\text{PL}}} \qquad (24) \\ &\propto -\frac{1}{N}\sum_{n=1}^{N}(f_{i\beta}(\hat{y}_{\beta,n}) + \frac{1}{|\beta|}\sum_{k\subset\beta}\mathbb{E}[f_{i\beta}(y_k,\hat{y}_{\beta\backslash k})]_{p_{k|-k}}) \end{aligned}$$

[7]The following considerations are easily generalized to include hidden variables, but for simplicity we have chosen to illustrate our point using observed variables only.

where $|\beta|$ denotes the number of nodes in the cluster β.

In light of our interpretation of learning in MRFs, it is not hard to generalize the PL estimator to a generalized PL estimator where we allow the relaxation of larger, possibly overlapping clusters of nodes conditioned on the remaining nodes in the graph. We leave the study of these generalized PL estimators as future work.

As mentioned above, it has been shown that the PL estimator is asymptotically consistent, but is less efficient than the ML estimator. It would be interesting to see if the arguments in the PL-consistency proofs can be adapted to cover the estimators studied in this paper.

6 CONDITIONAL RANDOM FIELDS

A conditional random field (CRF) [8] is a MRF that is trained to maximize the *conditional* log-likelihood of labels, y, given input variables x,

$$\lambda^{ML} = \arg\min_{\lambda} KL\left[P_0(y|x)||P_\lambda(y|x)\right] \qquad (25)$$

That is, the variables that appear in the data are partitioned into input nodes x, which will be observed at test time, and output nodes y, which we will be asked to predict at test time. In practice, discriminatively-trained models often have advantages over generatively-trained models, including the ability to include many interdependent variables in x without needing to learn their distribution.

All of our previous considerations apply to the conditional case as well. However, it should be noted that for generatively-trained models the free energy F_∞ must be computed with all the variables free to fluctuate. In contrast, for discriminatively-trained models the free energy F_∞ has the data-case x_n clamped to the input nodes. Hence, the learning rule aims to match the average sufficient statistics of the random system with 1) both x and y clamped at the nodes (F_0) and 2) the random system with only x clamped at the nodes (F_∞). This has the important consequence that the relaxed distributions $P_\lambda(y|x_n)$ are different for every data-case, while the relaxed distributions for generatively-trained models $P_\lambda(y)$ are the same for all data-cases and it would in principle suffice to run a single MCMC procedure per learning iteration [8].

7 EXPERIMENTS

In this section, we evaluate the $\mathbf{CF}_k$ estimators presented in this paper on CRFs. The state of the art for training loopy CRFs in practice is penalized maximum-likelihood training with the expected sufficient statistics computed by BP [12, 13]. This has two difficulties: (a) If the model distribution has multiple modes BP may converge to different solutions depending on its initialization (or fail to converge altogether), and (b) it requires running BP to convergence at each step of gradient ascent on the log-likelihood, which will be very expensive. Therefore, if nothing else, we can still hope to achieve improved training time by using the k-step $\mathbf{CF}$ estimators introduced in this paper. For the experiments in this paper, we will use fully-observed training data, leaving partially observed data to future work.

Method	F_1 (2-clique)	F_1 (4-clique)
$\mathbf{CF}_5^{\mathrm{BETHE}}$	**70.08**	74.94
$\mathbf{CF}_{10}^{\mathrm{BETHE}}$	68.35	75.23
$\mathbf{CF}_{15}^{\mathrm{BETHE}}$	61.80	76.51
$\mathbf{CF}_{500}^{\mathrm{BETHE}}$	63.44	75.86
$\mathbf{CF}_{10}^{\mathrm{MF}}$	57.91	55.91
$\mathrm{ML}^{\mathrm{MF}}$	60.98	65.31
$\mathrm{ML}^{\mathrm{BP}}$	68.19	**78.71**

Table 1: F_1 performance measure for various training methods on the 2-clique and 4-clique models.

Our data set is a collection of 485 e-mail messages announcing seminars at Carnegie Mellon University. The messages are annotated with the seminar's starting time, ending time, location, and speaker. This data set is due to Dayne Freitag [2], and has been used in much previous work. For reasons discussed in section 4, we consider here the binary problem of whether a word is a speaker name.

Often the speaker is listed multiple times in the same message. For example, the speaker's name might be included both near the beginning and later on, in a sentence like "If you would like to meet with Professor Smith…" It can be useful to find both such mentions, because different information can be in the surrounding context of each mention: for example, the first mention might be near an institution affiliation, while the second mentions that Smith is a professor.

To solve this problem, we wish to exploit that when the same word appears multiple times in the same message, it tends to have the same label. In a CRF, we can represent this by adding edges between output nodes (y_i, y_j) when the words x_i and x_j are identical and capitalized. Thus, the conditional distribution $p(y|x)$ has different graphical structure for different input configurations x. We use input nodes describing word identity, part-of-speech tags, capitalization, and membership in domain-specific lexicons; these are described in more detail elsewhere [11].

We compare training time and test performance of four dif-

[8]Note that we need to visit all modes with this Markov chain, so in practice it may be better to run multiple Markov chains initialized at various data-cases.

ferent contrastive free energies: $\mathrm{ML}^{\mathrm{MF}}$, which corresponds to maximum-likelihood training with mean-field free energy; $\mathrm{ML}^{\mathrm{BP}}$, which corresponds to maximum likelihood training with the Bethe free energy; and finally, $\mathbb{CF}_k^{\mathrm{MF}}$ and $\mathbb{CF}_k^{\mathrm{BETHE}}$, which correspond to k-step contrastive divergence with the mean-field and Bethe approximations, respectively. We compute the contrastive free energy as follows. For $\mathrm{ML}^{\mathrm{BP}}$, we use the TRP schedule for belief propagation [16], with messages initialized to 1. For $\mathrm{ML}^{\mathrm{MF}}$, we use damped fixed point equations with damping factor $\alpha = 0.1$ and uniform initialization. For $\mathbb{CF}_k^{\mathrm{MF}}$, however, we observed that iterating fixed-point equations for k steps might not decrease the free energy if they are improperly damped. Hence we have used separate damping factors for each data-case, $\alpha^{(i)}$, which are adapted to keep $\mathbb{CF}$ positive during learning.

To compute $\mathbb{CF}_k^{\mathrm{BETHE}}$, we use belief optimization; that is, we take k gradient steps on the Bethe free energy, eliminating the constraints by solving for the pairwise marginals and using the sigmoid parameterization described in [18]. The step-size for the gradient updates is determined by line search. For k-step contrastive divergence, it is essential that the optimization required to compute F_k^{BETHE} is initialized at the data cases. However, at the empirical distribution the derivative of the Bethe entropy is infinite. To avoid this problem we smooth the 0/1 empirical distribution by $\tilde{p}_{\mathrm{SOFT}}(x_j) = |\tilde{p}_{0/1}(x_j) - \epsilon|$. In these experiments we use $\epsilon = 10^{-4}$.

We report performance with the F_1 measure on a per-token basis, that is:

$$F_1 = (2PR)/(P + R) \tag{26}$$

with $P = \text{\# correct tokens} / \text{\# tokens extracted}$ and $R = \text{\# correct tokens} / \text{\# true tokens}$. We use ℓ_2 regularization with regularization parameter $\delta = 10$. All results are averaged over 5-fold cross validation.

First, we consider a *2-clique model* where all cliques are either linear chain edges (y_i, y_{i+1}), skip edges (y_i, y_j), and input edges (y_i, x_i)[9]. The parameters are tied over all instances of each clique type. For example, each linear chain edge (y_i, y_{i+1}) has the same weight w_{LC}. This sort of parameter tying is necessary in a conditional model because until we observe the input x, we do not know how many output nodes there will be or what connections they will have.

Table 1 compares the testing performance of the different training methods on the 2-clique model (first column). First, we note that both in $\mathbb{CF}$ and ML training, the Bethe approximation results in better accuracy than the mean-field approximation. This is as expected because the skip-chain model contains few short loops which is a graphical structure for which the Bethe approximation is more appropriate than the MF approximation. Second, with the Bethe free energy, using $\mathbb{CF}_5^{\mathrm{BETHE}}$ training results in comparable accuracy to ML training. This has great practical significance, because while the $\mathbb{CF}_5^{\mathrm{BETHE}}$ training used an average of 83 minutes to train, the ML training using belief propagation used over *19 hours*, which is an order of magnitude improvement.

Although the belief optimization algorithm has been developed for binary MRFs with pairwise interactions (a.k.a. Boltzmann machines), the CRF is free to contain arbitrary cliques with at most two output nodes, since the distribution $p(y|x)$ then still contains pairwise interactions only. To evaluate the practical advantages of such models, we also evaluate a skip chain model with higher-order cliques. In the 4-clique model, we add input nodes into the linear-chain and skip-chain cliques, so that we now have "linear-chain" cliques (y_i, y_{i+1}, x_i) and "skip" cliques (y_i, y_j, x_i, x_j) in addition to the input edges (y_i, x_i).

In Figure 1, we show the performance of $\mathbb{CF}_k^{\mathrm{BETHE}}$ model on the 4-clique model as a function of k (second column). For all values of k, the higher-order model performs better than the 2-clique model. Between the best 2-clique model and the best higher-order clique model, all 5 folds show improvement; averaging over the folds, the relative reduction in error is 20% (the F_1 rises from 70 to 76). For an unknown reason, the 2-clique model trained with $\mathbb{CF}_{15}^{\mathrm{BETHE}}$ hits a bad local maximum, but we do not see this behavior with a richer set of features. In the 4-clique model, ML training with BP does somewhat better than the best $\mathbb{CF}_k^{\mathrm{BETHE}}$ model, but there is substantial variance among the different training sets. None of the differences between $\mathrm{ML}^{\mathrm{BP}}$ and $\mathbb{CF}_k^{\mathrm{BETHE}}$ for the 4-clique model are statistically significant (McNemar's test with $p > 0.1$). For the 2-clique model, on the other hand, $\mathbb{CF}_5^{\mathrm{BETHE}}$ training is significantly better than $\mathrm{ML}^{\mathrm{BP}}$ ($p < 0.001$).

In summary, the experiments demonstrate two main points: that a k-step $\mathbb{CF}$ energy performs comparably to ML with vastly lower training time, and that belief optimization, which was developed for Boltzmann machines, is still effective for training models with certain higher-order cliques in a conditional setting.

8 CONCLUSION

In this paper we have offered a new view of parameter learning in MRF models as a minimization of contrastive free energies. We have seen that many objectives for MRF learning, including the likelihood function, the mean field learning objective, the contrastive divergence and the pseudo-likelihood can be written as a positive difference between two free energies. During learning we first infer the (posterior) distribution of the hidden variables given a

[9] To make the exposition simpler, we describe the models as if the only input variables x_i are the words at time i. In reality, each x_i is a vector of the observational tests described in [11].

clamped data-vector, then we relax this system (exactly, approximately or partially) by un-constraining the observed random variables. Finally we update the parameters by computing the difference of the average sufficient statistics. Not only is this unifying framework conceptually interesting, it also naturally suggests hybrid schemes where distributions are relaxed partially *and* approximately. In particular, we have studied a new learning algorithm based on the contrastive Kikuchi/Bethe free energy and its accompanying minimization algorithm, "belief optimization".

We feel that the view presented here is a rich breeding ground for new approximate learning algorithms. In future studies we hope to characterize the estimators proposed here by their asymptotic properties such as consistency and statistical efficiency.

Acknowledgments

This work was supported in part by the Center for Intelligent Information Retrieval and in part by The Central Intelligence Agency, the National Security Agency and National Science Foundation under NSF grant #IIS-0326249. Any opinions, findings and conclusions or recommendations expressed in this material are the author(s) and do not necessarily reflect those of the sponsor. Max Welling likes to thank G. Hinton, Y.W. Teh and S. Osindero for numerous discussions on the topic.

References

[1] J. Besag. Efficiency of pseudo-likelihood estimation for simple Gaussian fields. *Biometrika*, 64:616–618, 1977.

[2] Dayne Freitag. *Machine Learning for Information Extraction in Informal Domains*. PhD thesis, Carnegie Mellon University, 1998.

[3] B. Gidas. Consistency of maximum likelihood and pseudo-likelihood estimators for Gibbs distributions. In W. Fleming and eds. P.L. Lions, editors, *Stochastic Differential Systems, Stochastic Control Theory and Applications*. New York: Springer, 1988.

[4] T. Heskes. Stable fixed points of loopy belief propagation are minima of the Bethe free energy. In *Advances in Neural Information Processing Systems*, volume 15, Vancouver, CA, 2003.

[5] G.E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14:1771–1800, 2002.

[6] G.E. Hinton, K. Osindero, M. Welling, and Y.W. Teh. Unsupervised discovery of non-linear structure using contrastive backpropagation, 2004. in prepration.

[7] G.E. Hinton, M. Welling, and A. Mnih. Wormholes improve contrastive divergence. In *Advances in Neural Information Processing Systems*, volume 16, 2004.

[8] John Lafferty, Andrew McCallum, and Fernando Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. 18th International Conf. on Machine Learning*, pages 282–289. Morgan Kaufmann, San Francisco, CA, 2001.

[9] I. Murray and Z. Ghahramani. Bayesian learning in undirected graphical models: approximate MCMC algorithms. In *Proceedings of the 14th Annual Conference on Uncertainty in Artificial Intelligence (UAI-04)*, San Francisco, CA, 2004. Morgan Kaufmann Publishers.

[10] C. Peterson and J. Anderson. A mean field theory learning algorithm for neural networks. *Complex Systems*, 1:995–1019, 1987.

[11] Charles Sutton and Andrew McCallum. Collective segmentation and labeling of distant entities in information extraction. Technical Report TR # 04-49, University of Massachusetts, July 2004. Presented at ICML Workshop on Statistical Relational Learning and Its Connections to Other Fields.

[12] Charles Sutton, Khashayar Rohanimanesh, and Andrew McCallum. Dynamic conditional random fields: Factorized probabilistic models for labeling and segmenting sequence data. In *Proceedings of the Twenty-First International Conference on Machine Learning (ICML-2004)*, 2004.

[13] Ben Taskar, Pieter Abbeel, and Daphne Koller. Discriminative probabilistic models for relational data. In *Eighteenth Conference on Uncertainty in Artificial Intelligence (UAI02)*, 2002.

[14] Y.W. Teh and M. Welling. The unified propagation and scaling algorithm. In *Advances in Neural Information Processing Systems*, 2002.

[15] Y.W. Teh, M. Welling, S. Osindero, and G.E. Hinton. Energy-based models for sparse overcomplete representations. *Journal of Machine Learning Research - Special Issue on ICA*, 4:1235–1260, 2003.

[16] M.J. Wainwright, T. Jaakkola, and A.S. Willsky. Tree-based reparameterization for approximate estimation on loopy graphs. In *Advances Neural Information Processing Systems*, volume 14, vancouver, Canada, 2001.

[17] M. Welling and G.E. Hinton. A new learning algorithm for mean field Boltzmann machines. In *Proceedings of the International Conference on Artificial Neural Networks*, Madrid, Spain, 2001.

[18] M. Welling and Y.W. Teh. Belief optimization for binary networks: a stable alternative to loopy belief propagation. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, pages 554–561, Seattle, USA, 2001.

[19] J.S. Yedidia, W. Freeman, and Y. Weiss. Constructing free energy approximations and generalized belief propagation algorithms. Technical report, MERL, 2002. Technical Report TR-2002-35.

[20] A.L. Yuille. CCCP algorithms to minimize the Bethe and Kikuchi free energies: Convergent alternatives to belief propagation. *Neural Computation*, 14(7):1691–1722, 2002.

[21] A.L. Yuille. A comment on contrastive divergence. Technical report, Department Statistics and Psychology UCLA, 2004. Technical Report.