Variational Inference for Grammar Induction with Prior Knowledge

Shay B. Cohen and **Noah A. Smith**
Language Technologies Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213, USA
{scohen,nasmith}@cs.cmu.edu

Abstract

Variational EM has become a popular technique in probabilistic NLP with hidden variables. Commonly, for computational tractability, we make strong independence assumptions, such as the mean-field assumption, in approximating posterior distributions over hidden variables. We show how a looser restriction on the approximate posterior, requiring it to be a mixture, can help inject prior knowledge to exploit soft constraints during the variational E-step.

1 Introduction

Learning natural language in an unsupervised way commonly involves the expectation-maximization (EM) algorithm to optimize the parameters of a generative model, often a probabilistic grammar (Pereira and Schabes, 1992). Later approaches include variational EM in a Bayesian setting (Beal and Gharamani, 2003), which has been shown to obtain even better results for various natural language tasks over EM (e.g., Cohen et al., 2008).

Variational EM usually makes the mean-field assumption, factoring the posterior over hidden variables into independent distributions. Bishop et al. (1998) showed how to use a less strict assumption: a *mixture* of factorized distributions.

In other work, soft or hard constraints on the posterior during the E-step have been explored in order to improve performance. For example, Smith and Eisner (2006) have penalized the approximate posterior over dependency structures in a natural language grammar induction task to avoid long range dependencies between words. Graça et al. (2007) added linear constraints on expected values of features of the hidden variables in an alignment task.

In this paper, we use posterior mixtures to inject bias or prior knowledge into a Bayesian model. We show that empirically, injecting prior knowledge improves performance on an unsupervised Chinese grammar induction task.

2 Variational Mixtures with Constraints

Our EM variant encodes prior knowledge in an approximate posterior by constraining it to be from a *mixture* family of distributions. We will use $\mathbf{x}$ to denote observable random variables, $\mathbf{y}$ to denote hidden structure, and $\boldsymbol{\theta}$ to denote the to-be-learned parameters of the model (coming from a subset of $\mathbb{R}^\ell$ for some ℓ). $\boldsymbol{\alpha}$ will denote the parameters of a prior over $\boldsymbol{\theta}$. The mean-field assumption in the Bayesian setting assumes that the posterior has a factored form:

$$q(\boldsymbol{\theta}, \mathbf{y}) = q(\boldsymbol{\theta})q(\mathbf{y}) \tag{1}$$

Traditionally, variational inference with the mean-field assumption alternates between an E-step which optimizes $q(\mathbf{y})$ and then an M-step which optimizes $q(\boldsymbol{\theta})$.[1] The mean-field assumption makes inference feasible, at the expense of optimizing a looser lower bound on the likelihood (Bishop, 2006). The lower bound that the algorithm optimizes is the following:

$$F(q(\boldsymbol{\theta}, \mathbf{y}), \boldsymbol{\alpha}) = \mathbb{E}_{q(\boldsymbol{\theta},\mathbf{y})}[\log p(\mathbf{x}, \mathbf{y}, \boldsymbol{\theta} \mid \alpha)] + H(q) \tag{2}$$

where $H(q)$ denotes the entropy of distribution q. We focus on changing the E-step and as a result, changing the underlying bound, $F(q(\boldsymbol{\theta}, \mathbf{y}), \boldsymbol{\alpha})$. Similarly to Bishop et al. (1998), instead of making the strict mean-field assumption, we assume that the variational model is a mixture. One component of the mixture might take the traditional form, but others will be used to encourage certain

[1] This optimization can be nested inside another EM algorithm that optimizes $\boldsymbol{\alpha}$; this is our approach. $q(\boldsymbol{\theta})$ is traditionally conjugate to the likelihood for computational reasons, but our method is not limited to that kind of prior, as seen in the experiments.

Proceedings of the ACL-IJCNLP 2009 Conference Short Papers, pages 1–4,
Suntec, Singapore, 4 August 2009.

tendencies considered *a priori* to be appropriate. Denoting the probability simplex of dimension r $\triangle_r = \{\langle \lambda_1, ..., \lambda_r \rangle \in \mathbb{R}^r : \lambda_i \geq 0, \sum_{i=1}^r \lambda_i = 1\}$, we require that:

$$q(\boldsymbol{\theta}, \mathbf{y} \mid \boldsymbol{\lambda}) \quad = \quad \sum_{i=1}^r \lambda_i q_i(\mathbf{y}) q_i(\boldsymbol{\theta}) \qquad (3)$$

for $\boldsymbol{\lambda} \in \triangle_r$. $\mathcal{Q}_i$ will denote the family of distributions for the ith mixture component, and $\mathcal{Q}(\triangle_r)$ will denote the family implied by the mixture of $\mathcal{Q}_1, \ldots, \mathcal{Q}_r$ where the mixture coefficients $\boldsymbol{\lambda} \in \triangle_r$. $\boldsymbol{\lambda}$ comprise r additional variational parameters, in addition to parameters for each $q_i(\mathbf{y})$ and $q_i(\boldsymbol{\theta})$.

When one of the mixture components q_i is sufficiently expressive, $\boldsymbol{\lambda}$ will tend toward a degenerate solution. In order to force all mixture components to play a role—even at the expense of the tightness of the variational bound—we will impose hard constraints on $\boldsymbol{\lambda}$: $\boldsymbol{\lambda} \in \tilde{\triangle}_r \subset \triangle_r$. In our experiments (§3), $\tilde{\triangle}_r$ will be mostly a line segment corresponding to two mixture coefficients.

The role of the variational EM algorithm is to optimize the variational bound in Eq. 2 with respect to $q(\mathbf{y})$, $q(\boldsymbol{\theta})$, and $\boldsymbol{\lambda}$. Keeping this intention in mind, we can replace the E-step and M-step in the original variational EM algorithm with $2r + 1$ coordinate ascent steps, for $1 \leq i \leq r$:

E-step: For each $i \in \{1, ..., r\}$, optimize the bound given $\boldsymbol{\lambda}$ and $q_{i'}(\mathbf{y})|_{i' \in \{1,...,r\} \setminus \{i\}}$ and $q_{i'}(\boldsymbol{\theta})|_{i' \in \{1,...,r\}}$ by selecting a new distribution $q_i(\mathbf{y})$.

M-step: For each $i \in \{1, ..., r\}$, optimize the bound given $\boldsymbol{\lambda}$ and $q_{i'}(\boldsymbol{\theta})|_{i' \in \{1,...,r\} \setminus \{i\}}$ and $q_{i'}(\mathbf{y})|_{i' \in \{1,...,r\}}$ by selecting a new distribution $q_i(\boldsymbol{\theta})$.

C-step: Optimize the bound by selecting a new set of coefficients $\boldsymbol{\lambda} \in \tilde{\triangle}_r$ in order to optimize the bound with respect to the mixture coefficients.

We call the revised algorithm **constrained mixture variational EM**.

For a distribution $r(\mathbf{h})$, we denote by $\mathrm{KL}(\mathcal{Q}_i \| r)$ the following:

$$\mathrm{KL}(\mathcal{Q}_i \| r) = \min_{q \in \mathcal{Q}_i} \mathrm{KL}(q(\mathbf{h}) \| r)) \qquad (4)$$

where $\mathrm{KL}(\cdot \| \cdot)$ denotes the Kullback-Leibler divergence.

The next proposition, which is based on a result in Graça et al. (2007), gives an intuition of how modifying the variational EM algorithm with $\mathcal{Q} = \mathcal{Q}(\tilde{\triangle}_r)$ affects the solution:

Proposition 1. *Constrained mixture variational EM finds local maxima for a function $G(q, \boldsymbol{\alpha})$ such that*

$$\log p(x \mid \boldsymbol{\alpha}) - \min_{\boldsymbol{\lambda} \in \tilde{\triangle}_r} L(\boldsymbol{\lambda}, \boldsymbol{\alpha}) \leq G(q, \boldsymbol{\alpha}) \leq \log p(x \mid \boldsymbol{\alpha}) \qquad (5)$$

where $L(\boldsymbol{\lambda}, \boldsymbol{\alpha}) = \sum_{i=1}^r \lambda_i \mathrm{KL}(\mathcal{Q}_i \| p(\boldsymbol{\theta}, \mathbf{y} \mid \mathbf{x}, \boldsymbol{\alpha}))$.

We can understand mixture variational EM as penalizing the likelihood with a term bounded by a linear function of the $\boldsymbol{\lambda}$, minimized over $\tilde{\triangle}_r$. We will exploit that bound in §2.2 for computational tractability.

2.1 Simplex Annealing

The variational EM algorithm still identifies only *local* maxima. Different proposals have been for pushing EM toward a global maximum. In many cases, these methods are based on choosing different initializations for the EM algorithm (e.g., repeated random initializations or a single carefully designed initializer) such that it eventually gets closer to a global maximum.

We follow the idea of annealing proposed in Rose et al. (1990) and Smith and Eisner (2006) for the $\boldsymbol{\lambda}$ by gradually loosening hard constraints on $\boldsymbol{\lambda}$ as the variational EM algorithm proceeds. We define a sequence of $\tilde{\triangle}_r(t)$ for $t = 0, 1, ...$ such that $\tilde{\triangle}_r(t) \subseteq \tilde{\triangle}_r(t+1)$. First, we have the inequality:

$$\begin{aligned} &\mathrm{KL}(\mathcal{Q}(\tilde{\triangle}_r(t)) \| p(\boldsymbol{\theta}, \mathbf{y} \mid \mathbf{x}, \boldsymbol{\alpha}) \\ &\quad \geq \quad \mathrm{KL}(\mathcal{Q}(\tilde{\triangle}_r(t+1)) \| p(\boldsymbol{\theta}, \mathbf{y} \mid \mathbf{x}, \boldsymbol{\alpha})) \end{aligned} \qquad (6)$$

We say that the annealing schedule is τ-separated if we have for any $\boldsymbol{\alpha}$:

$$\begin{aligned} &\mathrm{KL}(\mathcal{Q}(\tilde{\triangle}_r(t)) \| p(\boldsymbol{\theta}, \mathbf{y} \mid \mathbf{x}, \boldsymbol{\alpha})) \\ &\quad \leq \quad \mathrm{KL}(\mathcal{Q}(\tilde{\triangle}_r(t+1)) \| p(\boldsymbol{\theta}, \mathbf{y} \mid \mathbf{x}, \boldsymbol{\alpha})) - \frac{\tau}{2^{(t+1)}} \end{aligned} \qquad (7)$$

τ-separation requires consecutive families $\mathcal{Q}(\tilde{\triangle}_r(t))$ and $\mathcal{Q}(\tilde{\triangle}_r(t+1))$ to be similar.

Proposition 1 stated the bound we optimize, which penalizes the likelihood by subtracting a positive KL divergence from it. With the τ-separation condition we can show that even though we penalize likelihood, the variational EM algorithm will still increase likelihood by a certain amount. Full details are omitted for space and can be found in **?**).

Input: initial parameters $\boldsymbol{\alpha}^{(0)}$, observed data $\mathbf{x}$, annealing schedule $\tilde{\triangle}_r : \mathbb{N} \rightarrow 2^{\triangle_r}$
Output: learned parameters $\boldsymbol{\alpha}$ and approximate posterior $q(\boldsymbol{\theta}, \mathbf{y})$

$t \leftarrow 1$;
repeat
  E-step: **repeat**
    E-step: **forall** $i \in [r]$ do: $q_i^{(t+1)}(\mathbf{y}) \leftarrow \operatorname{argmax}_{q(\mathbf{y}) \in \mathcal{Q}_i} F'(\sum_{j \neq i} \lambda_j q_i^{(t)}(\boldsymbol{\theta}) q(\mathbf{y}) + \lambda_i q_i^{(t)} q(\mathbf{y}), \boldsymbol{\alpha}^{(t)})$
    M-step: **forall** $i \in [r]$ do: $q_i^{(t+1)}(\boldsymbol{\theta}) \leftarrow \operatorname{argmax}_{q(\boldsymbol{\theta}) \in \mathcal{Q}_i} F'(\sum_{j \neq i} \lambda_j q(\boldsymbol{\theta}) q_i^{(t)}(\mathbf{y}) + \lambda_i q_i^{(t)} q(\mathbf{y}), \boldsymbol{\alpha}^{(t)})$
    C-step: $\boldsymbol{\lambda}^{(t+1)} \leftarrow \operatorname{argmax}_{\boldsymbol{\lambda} \in \tilde{\triangle}_r(t)} F'(\sum_{j=1}^{r} \lambda_j q_i^{(t)}(\boldsymbol{\theta}) q_i^{(t)}(\mathbf{y}), \boldsymbol{\alpha}^{(t)})$
  until *convergence* ;
  M-step: $\boldsymbol{\alpha}^{(t+1)} \leftarrow \operatorname{argmax}_{\boldsymbol{\alpha}} F'(\sum_{i=1}^{r} \lambda_i q_i^{(t+1)}(\boldsymbol{\theta}) q_i^{(t+1)}(\mathbf{y}), \boldsymbol{\alpha})$
  $t \leftarrow t + 1$;
until *convergence* ;
return $\boldsymbol{\alpha}^{(t)}, \sum_{i=1}^{r} \lambda_i q_i^{(t)}(\boldsymbol{\theta}) q_i^{(t)}(\mathbf{y})$

Figure 1: The constrained variational mixture EM algorithm. $[n]$ denotes $\{1, ..., n\}$.

2.2 Tractability

We now turn to further alterations of the bound in Eq. 2 to make it more tractable. The main problem is the entropy term which is not easy to compute, because it includes a $\log$ term over a mixture of distributions from $\mathcal{Q}_i$. We require the distributions in $\mathcal{Q}_i$ to factorize over the hidden structure $\mathbf{y}$, but this only helps with the first term in Eq. 2.

We note that because the entropy function is convex, we can get a lower bound on $H(q)$:

$$H(q) \geq \sum_{i=1}^{r} \lambda_i H(q_i) = \sum_{i=1}^{r} \lambda_i H(q_i(\boldsymbol{\theta}, \mathbf{y}))$$

Substituting the modified entropy term into Eq. 2 still yields a lower bound on the likelihood. This change makes the E-step tractable, because each distribution $q_i(\mathbf{y})$ can be computed separately by optimizing a bound which depends only on the variational parameters in that distribution. In fact, the bound on the left hand side in Proposition 1 becomes the function that we optimize instead of $G(q, \boldsymbol{\alpha})$.

Without proper constraints, the $\boldsymbol{\lambda}$ update can be intractable as well. It requires maximizing a linear objective (in $\boldsymbol{\lambda}$) while constraining the $\boldsymbol{\lambda}$ to be from a particular subspace of the probability simplex, $\tilde{\triangle}_r(t)$. To solve this issue, we require that $\tilde{\triangle}_r(t)$ is polyhedral, making it possible to apply linear programming (Boyd and Vandenberghe, 2004).

The bound we optimize is:[2]

$$F'\left(\sum_{i=1}^{r} \lambda_i q_i(\boldsymbol{\theta}, \mathbf{y}), \boldsymbol{\alpha}\right) \tag{8}$$
$$= \sum_{i=1}^{r} \lambda_i \left(\mathbb{E}_{q_i(\boldsymbol{\theta}, \mathbf{y})}[\log p(\boldsymbol{\theta}, \mathbf{y}, \mathbf{x} \mid \mathbf{m})] + H(q_i(\boldsymbol{\theta}, \mathbf{y}))\right)$$

with $\boldsymbol{\lambda} \in \tilde{\triangle}_r(t_{\text{final}})$ and $(q_i(\boldsymbol{\theta}, \mathbf{y})) \in \mathcal{Q}_i$. The algorithm for optimizing this bound is in Fig. 1, which includes an extra M-step to optimize $\boldsymbol{\alpha}$ (see extended report).

3 Experiments

We tested our method on the unsupervised learning problem of dependency grammar induction. For the generative model, we used the *dependency model with valence* as it appears in Klein and Manning (2004). We used the data from the Chinese treebank (Xue et al., 2004). Following standard practice, sentences were stripped of words and punctuation, leaving part-of-speech tags for the unsupervised induction of dependency structure, and sentences of length more than 10 were removed from the set. We experimented with a Dirichlet prior over the parameters and logistic normal priors over the parameters, and found the latter to still be favorable with our method, as in Cohen et al. (2008). We therefore report results with our method only for the logistic normal prior. We do inference on sections 1–270 and 301–1151 of CTB10 (4,909 sentences) by running the EM algorithm for 20 iterations, for which all algorithms have their variational bound converge.

To evaluate performance, we report the fraction of words whose predicted parent matches the gold standard (attachment accuracy). For parsing, we use the minimum Bayes risk parse.

Our mixture components $\mathcal{Q}_i$ are based on simple linguistic tendencies of Chinese syntax. These observations include the tendency of dependencies to (a) emanate from the right of the current position and (b) connect words which are nearby (in string distance). We experiment with six mixture components: (1) RIGHTATTACH: Each word's parent is to the word's right. The root, therefore, is always the rightmost word; (2) ALLRIGHT: The rightmost word is the parent of all positions in the sentence (there is only one such tree); (3) LEFTCHAIN: The tree forms a chain, such that each

[2] This is a less tight bound than the one in Bishop et al. (1998), but it is easier to handle computationally.

learning setting				
	LEFTCHAIN	34.9		
	vanilla EM	38.3		
	LN, mean-field	48.9		
	This paper:	I	II	III
	RIGHTATTACH	49.1	47.1	**49.8**
	ALLRIGHT	**49.4**	**49.4**	48.4
	LEFTCHAIN	47.9	46.5	**49.9**
	VERBASROOT	**50.5**	50.2	49.4
	NOUNSEQUENCE	48.9	48.9	**49.9**
	SHORTDEP	**49.5**	48.4	48.4
	RA+VAR+SD	50.5	**50.6**	50.1

Table 1: Results (attachment accuracy). The baselines are LEFTCHAIN as a parsing model (attaches each word to the word on its right), non-Bayesian EM, and mean-field variational EM without any constraints. These are compared against the six mixture components mentioned in the text. (I) corresponds to simplex annealing experiments ($\lambda_1^{(0)} = 0.85$); (II–III) correspond to fixed values, 0.85 and 0.95, for the mixture coefficients. With the last row, λ_2 to λ_4 are always $(1 - \lambda_1)/3$. Boldface denotes the best result in each row.

word is governed by the word to its right; (4) VERBASROOT: Only verbs can attach to the wall node \$; (5) NOUNSEQUENCE: Every sequence of n NN (nouns) is assumed to be a noun phrase, hence the first $n-1$ NNs are attached to the last NN; and (6) SHORTDEP: Allow only dependencies of length four or less. This is a strict model reminiscent of the successful application of *structural bias* to grammar induction (Smith and Eisner, 2006).

These components are added to a variational DMV model without the sum-to-1 constraint on $\boldsymbol{\theta}$. This complements variational techniques which state that the optimal solution during the E-step for the mean-field variational EM algorithm is a weighted grammar of the same form of $p(\mathbf{x}, \mathbf{y} \mid \boldsymbol{\theta})$ (DMV in our case). Using the mixture components this way has the effect of *smoothing* the estimated grammar event counts during the E-step, in the direction of some prior expectations.

Let λ_1 correspond to the component of the original DMV model, and let λ_2 correspond to one of the components from the above list. Variational techniques show that if we let λ_1 obtain the value 1, then the optimal solution will be $\lambda_1 = 1$ and $\lambda_2 = 0$. We therefore restrict λ_1 to be smaller than 1. More specifically, we use an annealing process which starts by limiting λ_1 to be $\leq s = 0.85$ (and hence limits λ_2 to be ≥ 0.15) and increases s at each step by 1% until s reaches 0.95. In addition, we also ran the algorithm with λ_1 fixed at 0.85 and λ_1 fixed at 0.95 to check the effectiveness of annealing on the simplex.

Table 1 describes the results of our experiments. In general, using additional mixture components has a clear advantage over the mean-field assumption. The best result with a single mixture is achieved with annealing, and the VERBASROOT component. A combination of the mixtures (RIGHTATTACH) together with VERBASROOT and SHORTDEP led to an additional improvement, implying that proper selection of several mixture components together can achieve a performance gain.

4 Conclusion

We described a variational EM algorithm that uses a mixture model for the variational model. We refined the algorithm with an annealing mechanism to avoid local maxima. We demonstrated the effectiveness of the algorithm on a dependency grammar induction task. Our results show that with a good choice of mixture components and annealing schedule, we achieve improvements for this task over mean-field variational inference.

References

M. J. Beal and Z. Gharamani. 2003. The variational Bayesian EM algorithm for incomplete data: with application to scoring graphical model structures. In *Proc. of Bayesian Statistics*.

C. Bishop, N. Lawrence, T. S. Jaakkola, and M. I. Jordan. 1998. Approximating posterior distributions in belief networks using mixtures. In *Advances in NIPS*.

C. M. Bishop. 2006. *Pattern Recognition and Machine Learning*. Springer.

S. Boyd and L. Vandenberghe. 2004. *Convex Optimization*. Cambridge Press.

S. B. Cohen and N. A. Smith. 2009. Variational inference with prior knowledge. Technical report, Carnegie Mellon University.

S. B. Cohen, K. Gimpel, and N. A. Smith. 2008. Logistic normal priors for unsupervised probabilistic grammar induction. In *Advances in NIPS*.

J. V. Graça, K. Ganchev, and B. Taskar. 2007. Expectation maximization and posterior constraints. In *Advances in NIPS*.

D. Klein and C. D. Manning. 2004. Corpus-based induction of syntactic structure: Models of dependency and constituency. In *Proc. of ACL*.

F. C. N. Pereira and Y. Schabes. 1992. Inside-outside reestimation from partially bracketed corpora. In *Proc. of ACL*.

K. Rose, E. Gurewitz, and G. C. Fox. 1990. Statistical mechanics and phrase transitions in clustering. *Physical Review Letters*, 65(8):945–948.

N. A. Smith and J. Eisner. 2006. Annealing structural bias in multilingual weighted grammar induction. In *Proc. of COLING-ACL*.

N. Xue, F. Xia, F.-D. Chiou, and M. Palmer. 2004. The Penn Chinese Treebank: Phrase structure annotation of a large corpus. *Natural Language Engineering*, 10(4):1–30.