An Application of the Variational Bayesian Approach to Probabilistic Context-Free Grammars

Kenichi Kurihara
Department of Computer Science
Tokyo Institute of Technology, Tokyo
kurihara@mi.cs.titech.ac.jp

Taisuke Sato
Department of Computer Science
Tokyo Institute of Technology, Tokyo
sato@mi.cs.titech.ac.jp

Abstract

We present an efficient learning algorithm for probabilistic context-free grammars based on the variational Bayesian approach. Although the maximum likelihood method has traditionally been used for learning probabilistic language models, Bayesian learning is, in principle, less likely to cause overfitting problems than the maximum likelihood method. We show that the computational complexity of our algorithm is equal to that of the Inside-Outside algorithm. We also report results of experiments to compare precisions of the Inside-Outside algorithm and our algorithm.

1 Introduction

In natural language processing, one of the main problems is a reduction of syntactic ambiguity of sentences. Generally, a sentence can have so many syntactic structures that we need to decide which structure is the most likely. Probabilistic language models have been used to solve this problem. The maximum likelihood method is well-known as a learning algorithm for parameters of probabilistic models. Especially, the expectation maximization (EM) algorithm, a class of the maximum likelihood method, has been developed for learning with incomplete data. For example, Baum-Welch algorithm for hidden Markov models and the Inside-Outside algorithm for probabilistic context-free grammars (PCFGs) are well-known. However, the maximum likelihood method is known to be likely to cause overfitting problems. Recently, it has been showed that Bayesian learning is theoretically more reliable than the maximum likelihood method in some models (Hartigan, 1985; Watanabe, 2001). As an efficient approximation of Bayesian learning, the variational Bayesian approach has been developed (Attias, 1999).

Although the variational Bayesian approach looks promising, the variational Bayesian approach has not been applied to PCFGs to our knowledge. In this paper, we apply the variational Bayesian approach to PCFGs and derive an efficient learning algorithm based on dynamic programming. We show that the computational complexity of our algorithm is equal to that of the Inside-Outside algorithm. We also report the results of comparing our algorithm with the Inside-Outside algorithm.

2 Probabilistic Context-Free Grammar

In this section, we define some notation. Let $G = (V_N, V_T, R, S)$ be a PCFG, where V_N, V_T and R are respectively a set of non-terminals, terminals and rules, and S is a start symbol. $\boldsymbol{\theta}(r)$ $(r \in R)$ is a parameter of r. Especially, $\boldsymbol{\theta}_A(\alpha)$ stands for $\boldsymbol{\theta}(A \rightarrow \alpha)$ $(A \in V_N, \alpha \in (V_N \cup V_T)^+)$. We assume that a prior probability over the parameters $\boldsymbol{\theta}$ is a product of Dirichlet distributions as follows

$$p(\boldsymbol{\theta}) = \prod_{A \in V_N} P_D(\boldsymbol{\theta}_A, \boldsymbol{u}_A), \tag{1}$$

where $\boldsymbol{u}_A$ is a vector of the hyperparameters of $\boldsymbol{\theta}_A$:

$$\boldsymbol{u}_A = \{u_{A \rightarrow \alpha} | A \rightarrow \alpha \in R\}. \tag{2}$$

Dirichlet distribution P_D is defined by

$$P_D(\boldsymbol{\theta}_A, \boldsymbol{u}_A) = \frac{1}{Z} \prod_{\alpha; A\to\alpha \in R} \boldsymbol{\theta}_A(\alpha)^{u_{A\to\alpha}-1}, \quad (3)$$

$$Z = \frac{\prod_{\alpha; A\to\alpha\in R} \Gamma(u_{A\to\alpha})}{\Gamma\left(\sum_{\alpha; A\to\alpha\in R} u_{A\to\alpha}\right)}, \quad (4)$$

where Γ is the gamma function.

3 Variational Bayesian Approach for Probabilistic Context-Free Grammar

While the maximum likelihood method optimize a point estimate of parameters, Bayesian learning estimates a posterior distribution of the parameters $p(\boldsymbol{\theta}|C)$. Since Bayesian learning is difficult due to its intractable integration, the variational Bayesian approach assumes simplifying constraints to estimate an approximate posterior distribution $q(\boldsymbol{\theta}|C)$ efficiently.

3.1 Posterior Distribution

In this section, we derive a basic learning algorithm for PCFGs to calculate the approximate posterior distribution $q(\boldsymbol{\theta}|C)$.

Let C be a training corpus and $\mathcal{R}$ be a set of derivation sequences, and put

$$C = (s_1, s_2, \cdots, s_N), \quad \mathcal{R} = (\boldsymbol{r}_1, \boldsymbol{r}_2, \cdots, \boldsymbol{r}_N),$$

where $\boldsymbol{r}_i$ is a derivation sequence of a sentence s_i. We define $\mathcal{F}$ using Jensen's inequality as follows

$$\begin{aligned} \mathcal{L}(C) &= \log p(C) \\ &= \log \sum_{\mathcal{R}} \int p(C, \mathcal{R}, \boldsymbol{\theta}) d\boldsymbol{\theta} \\ &= \log \sum_{\mathcal{R}} \int q(\mathcal{R}, \boldsymbol{\theta}|C) \frac{p(C, \mathcal{R}, \boldsymbol{\theta})}{q(\mathcal{R}, \boldsymbol{\theta}|C)} d\boldsymbol{\theta} \\ &\geq \sum_{\mathcal{R}} \int q(\mathcal{R}, \boldsymbol{\theta}|C) \log \frac{p(C, \mathcal{R}, \boldsymbol{\theta})}{q(\mathcal{R}, \boldsymbol{\theta}|C)} d\boldsymbol{\theta} = \mathcal{F} \end{aligned}$$

where $\mathcal{L}(C)$ is a log likelihood of the training corpus C, $p(\mathcal{R}, \boldsymbol{\theta}|C)$ is a posterior distribution and $q(\mathcal{R}, \boldsymbol{\theta}|C)$ is a free distribution.

We note that the difference between $\mathcal{L}(C)$ and $\mathcal{F}$ is Kullback-Leibler (KL) distance as follows

$$\begin{aligned} \mathcal{L}(C) - \mathcal{F} &= \sum_{\mathcal{R}} \int q(\mathcal{R}, \boldsymbol{\theta}|C) \log \frac{q(\mathcal{R}, \boldsymbol{\theta}|C)}{p(\mathcal{R}, \boldsymbol{\theta}|C)} d\boldsymbol{\theta} \\ &= KL(q(\mathcal{R}, \boldsymbol{\theta}|C), p(\mathcal{R}, \boldsymbol{\theta}|C)). \quad (5) \end{aligned}$$

Since $\mathcal{L}(C)$ is constant when C is fixed, maximizing $\mathcal{F}$ is equivalent to minimizing KL distance between $q(\mathcal{R}, \boldsymbol{\theta}|C)$ and $p(\mathcal{R}, \boldsymbol{\theta}|C)$. Therefore, we can obtain an approximate posterior distribution $q(\mathcal{R}, \boldsymbol{\theta}|C)$ by maximizing $\mathcal{F}$ as a functional of $q(\mathcal{R}, \boldsymbol{\theta}|C)$.

Here, we constrain that $q(\boldsymbol{\theta}|C)$ and $q(\mathcal{R}|C)$ are independent such that

$$\begin{aligned} q(\mathcal{R}, \boldsymbol{\theta}|C) &= q(\mathcal{R}|C) q(\boldsymbol{\theta}|C) \\ &= \left\{ \prod_{i=1}^{N} q(\boldsymbol{r}|s_i) \right\} \left\{ \prod_{A \in V_N} q(\boldsymbol{\theta}_A|C) \right\}. \end{aligned}$$

By maximizing $\mathcal{F}$ as a functional of $q(\boldsymbol{\theta}|C)$ with $q(\mathcal{R}|C)$ fixed, $q(\boldsymbol{\theta}|C)$ is obtained as

$$q(\boldsymbol{\theta}|C) = \prod_{A \in V_N} P_D(\boldsymbol{\theta}_A, \hat{\boldsymbol{u}}_A) \quad (6)$$

where

$$\hat{\boldsymbol{u}}_A = \{\hat{u}_{A\to\alpha} | A \to \alpha \in R\}, \quad (7)$$

$$\hat{u}_r = u_r + \sum_{i=1}^{N} \sum_{\boldsymbol{r} \in \Phi(s_i)} q(\boldsymbol{r}|s_i) c(r; \boldsymbol{r}), \quad (8)$$

u_r is a hyperparameter of the prior distribution, $c(r; \boldsymbol{r})$ is the number of occurrences of the rule r in $\boldsymbol{r}$ and $\Phi(s_i)$ is a set of derivation sequences which derive s_i.

Similarly, $q(\mathcal{R}|C)$ is calculated as

$$\begin{aligned} q(\mathcal{R}|C) &= \prod_{i=1}^{N} q(\boldsymbol{r}|s_i) \\ &= \prod_{i=1}^{N} \frac{\prod_{r \in R} \pi(r)^{c(r;\boldsymbol{r})}}{\sum_{\boldsymbol{r} \in \Phi(s_i)} \prod_{r \in R} \pi(r)^{c(r;\boldsymbol{r})}}, \quad (9) \end{aligned}$$

where

$$\pi(A \to \alpha) = \exp\left[\psi(\hat{u}_{A\to\alpha}) - \psi\left(\sum_{\alpha; A\to\alpha \in R} \hat{u}_{A\to\alpha} \right) \right],$$

and ψ is the digamma function.

Since $q(\boldsymbol{\theta}|C)$ and $q(\mathcal{R}|C)$ depend on each other, the optimal distribution $q(\boldsymbol{\theta}|C)$ can be computed by updating $q(\boldsymbol{\theta}|C)$ and $q(\mathcal{R}|C)$ alternately with equations (6) and (9). During updating equations (6) and (9), $\mathcal{F}$ increases monotonically. Finally, equations for updating hyperparameters $\boldsymbol{u}$ for $q(\boldsymbol{\theta}|C)$ are expressed as follows.

$$u_r^{(0)} = u_r$$

$$u_r^{(k+1)} = u_r + \sum_{i=1}^{N} \sum_{\boldsymbol{r}\in\Phi(s_i)} q^{(k)}(\boldsymbol{r}|s_i)c(r;\boldsymbol{r}) \tag{10}$$

$$q^{(k)}(\boldsymbol{r}|s_i) = \frac{\prod_{r\in R}\pi^{(k)}(r)^{c(r;\boldsymbol{r})}}{\sum_{\boldsymbol{r}\in\Phi(s_i)}\prod_{r\in R}\pi^{(k)}(r)^{c(r;\boldsymbol{r})}}$$

$$\pi^{(k)}(A\rightarrow\alpha) = \exp\left[\psi(u^{(k)}_{A\rightarrow\alpha}) - \psi\left(\sum_{\alpha;A\rightarrow\alpha\in R} u^{(k)}_{A\rightarrow\alpha}\right)\right]$$

The optimal distribution $q^*(\boldsymbol{\theta}|C)$ is obtained as

$$q^*(\boldsymbol{\theta}|C) = \prod_{A\in V_N} P_D(\boldsymbol{\theta}_A, \boldsymbol{u}^*_A), \tag{11}$$

where $\boldsymbol{u}^*$ is the convergent value of $\boldsymbol{u}^{(k)}$.

3.2 Calculation of Hyperparameters

Since the size of $\Phi(s_i)$ is $O(\exp(l_i))$ where l_i is the length of s_i, equation (10) requires exponential time. To reduce this computational complexity, we apply dynamic programming to equation (10).

3.2.1 Inside-Outside Algorithm

First, we briefly review the Inside-Outside algorithm which is the first EM algorithm for PCFGs based on dynamic programming (Baker, 1979). The Inside-Outside algorithm estimates optimal parameters $\boldsymbol{\theta}$, which are updated iteratively by the following equation

$$\hat{\boldsymbol{\theta}}(A\rightarrow\alpha) = \frac{1}{Z_A}\sum_{i=1}^{N} c(A\rightarrow\alpha; s_i) \tag{12}$$

where

$$\begin{aligned} c(r;s_i) &= \sum_{\boldsymbol{r}\in\Phi(s_i)} p(\boldsymbol{r}|s_i,\boldsymbol{\theta})c(r;\boldsymbol{r}) \\ &= \sum_{\boldsymbol{r}\in\Phi(s_i)} \frac{\prod_{r\in R}\boldsymbol{\theta}(r)^{c(r;\boldsymbol{r})}}{\sum_{\boldsymbol{r}\in\Phi(s_i)}\prod_{r\in R}\boldsymbol{\theta}(r)^{c(r;\boldsymbol{r})}} c(r;\boldsymbol{r}). \end{aligned} \tag{13}$$

We assume that the grammar is in Chomsky normal form. Thus, each rule is either of the form $A\rightarrow BC$ or $A\rightarrow a$ ($A, B, C \in V_N$, $a \in V_T$). Let $s = (w_1, w_2, \cdots, w_l)$ and $w_i^j = (w_i, w_{i+1}, \cdots, w_j)$. We define the outside probability α and the inside probability β as follows

$$\alpha_{i,j}(A) = p(S \overset{*}{\Rightarrow} w_1^{i-1}Aw_{j+1}^{l}) \tag{14}$$

$$\beta_{i,j}(A) = p(A \overset{*}{\Rightarrow} w_i^j). \tag{15}$$

Finally, $c(r;s)$ can be calculated efficiently as follows

$$\begin{aligned} c(A\rightarrow BC; s) &= \frac{\boldsymbol{\theta}(A\rightarrow BC)}{p(s|\boldsymbol{\theta})} \\ &\times \sum_{n=1}^{l-1}\sum_{i=1}^{l-n}\sum_{j=1}^{n} \alpha_{i,i+n}(A)\beta_{i,i+j-1}(B)\beta_{i+j,i+n}(C), \end{aligned}$$

$$c(A\rightarrow a; s) = \frac{\boldsymbol{\theta}(A\rightarrow a)}{p(s|\boldsymbol{\theta})}\sum_{n=1}^{l}\alpha_{i,i}(A),$$

where $A, B, C \in V_N$ and $a \in V_T$ (Lafferty, 1993).

3.2.2 Efficient Estimation of Hyperparameters

By defining $\gamma(r;s)$ like $c(r;s)$ in the Inside-Outside algorithm,

$$\begin{aligned} \gamma^{(k)}(r;s_i) &= \sum_{\boldsymbol{r}\in\Phi(s_i)} q^{(k)}(\boldsymbol{r}|s_i)c(r;\boldsymbol{r}), \\ &= \sum_{\boldsymbol{r}\in\Phi(s_i)} \frac{\prod_{r\in R}\pi^{(k)}(r)^{c(r;\boldsymbol{r})}}{\sum_{\boldsymbol{r}\in\Phi(s_i)}\prod_{r\in R}\pi^{(k)}(r)^{c(r;\boldsymbol{r})}} c(r;\boldsymbol{r}), \end{aligned} \tag{16}$$

equation (10) is rewritten as

$$u_r^{(k+1)} = u_r + \sum_{i=1}^{N}\gamma^{(k)}(r;s_i). \tag{17}$$

Since equation (16) is just equation (13) with $\boldsymbol{\theta}(r)$ replaced by $\pi^{(k)}(r)$, equation (16) can be calculated efficiently with the Inside-Outside algorithm.

3.3 Predictive Posterior Distribution

Once we have obtained the optimal posterior distribution $q^*(\boldsymbol{\theta}|C)$, we can predict the most likely derivation sequence of an unknown sentence s using a predictive posterior distribution of $\boldsymbol{r}$ defined by

$$\begin{aligned} p(\boldsymbol{r}|s,C) &= \int p(\boldsymbol{r}|s,\boldsymbol{\theta})q^*(\boldsymbol{\theta}|C)d\boldsymbol{\theta} \\ &\propto \frac{\prod_{r\in R}\Gamma(u_r^* + c(r;\boldsymbol{r}))}{\prod_{A\in V_N}\Gamma(\sum_{\alpha;A\rightarrow\alpha\in R}u^*_{A\rightarrow\alpha} + c(A\rightarrow\alpha;\boldsymbol{r}))}. \end{aligned} \tag{18}$$

Table 1: Evaluation of the Inside-Outside algorithm and the proposed algorithm with 2,199 sentences and 8,796 sentences as training corpora

2,199 sentences	LP	Exact	0 CB
Inside-Outside	95.05	75.24	87.03
proposed algorithm	95.89	76.73	89.13

8,796 sentences	LP	Exact	0 CB
Inside-Outside	96.19	77.39	89.18
proposed algorithm	96.25	77.41	89.37

Since we can't apply a Viterbi-style algorithm to equation (18), we need to calculate equation (18) for all the derivations in order to decide the most likely derivation.

4 Experiments

We conducted learning experiments to compare our algorithm with the Inside-Outside algorithm. The used training corpus was ATR corpus (Uratani, 1994), which contains labeled 10,995 sentences and a context-free grammar with 861 rules. The mean value of the length of the sentences is 9.97. In this study, we converted the grammar to Chomsky normal form. The converted grammar has 2,336 rules, 226 non-terminals and 441 terminals. The training data has no labels or brackets. We performed a five fold cross validation and evaluated the results based on labeled precision (LP), exact match and zero crossing brackets (0 CB). Labeled precision is equal to labeled recall, since the grammar we used is in Chomsky normal form. For the Inside-Outside algorithm, we set initial parameters to an uniform distribution. For our algorithm, $u_r = 2$ ($\forall r \in R$) was used.

We used two sets of test and training corpora. One set contained 2,199 sentences for training and 8,796 sentences for the test. The other set contained 8,796 sentences for training and 2,199 sentences for the test.

Table 1 shows the evaluations of the predictions. In this figure, our algorithm achieves better predictive accuracy than the EM algorithm. Especially, the differences of two algorithms are slightly larger with 2,199 training sentences than with 8,796 sentences. These results imply that the variational Bayesian approach is more reliable than the EM algorithm especially with small training corpora.

5 Related Work

MacKay has applied the variational Bayesian approach to hidden Markov models and derived an algorithm whose computational complexity is equal to that of Baum-Welch algorithm (MacKay, 1997). He has also pointed out difficulty in finding the most likely derivation as we mentioned in section 3.3.

6 Conclusion

We applied the variational Bayesian approach to parameter learning of PCFGs and derived an learning algorithm. In addition, We improved the algorithm using dynamic programming so that the computational complexity of the improved algorithm is reduced to that of the Inside-Outside algorithm. The experiments show the variational Bayesian approach is better than the EM algorithm.

As the future work, it is required to overcome computational difficulty in prediction of the most likely derivation. Currently, prediction is a computationally intractable task as the Viterbi-style algorithm is not applicable.

We also need to evaluate the performance of the variational Bayesian approach for other types of corpora, especially partially bracketed corpora.

Acknowledgments

This research was funded by the 21st Century COE Program "Framework for Systematization and Application of Large-scale Knowledge Resources."

References

Hagai Attias. 1999. *Inferring Parameters and Structure of Latent Variable Models by Variational Bayes*. Proc. Uncertainty in Artificial Intelligence.

James K. Baker. 1979. *Trainable grammars for speech recognition*. In Jared J. Wolf and Dennis H. Klatt, editors, Speech communication papers presented at the 97th Meeting of the Acoustical Society of America, 547–550.

J A. Hartigan. 1985. *A failure of likelihood asymptotics for normal mixtures*. Proceedings of the Berkeley Conference in Honor of J.Neyman and J.Kiefer, 2, 807–810.

John D. Lafferty. 1993. *A derivation of the Inside-Outside algorithm from the EM algorithm*. IBM T.J. Watson Research Center.

David J.C. MacKay. 1997. *Ensemble Learning for Hidden Markov Models*. Technical Report, University of Cambridge.

Noriyoshi Uratani, Toshiyuki Takezawa, Hidehiko Matsuo, and Chiho Morita. 1994. *ATR Integrated Speech and Language Database*. Technical Report TR-IT-0056, ATR Interpreting Telecommunications Research Laboratories.

Sumio Watanabe. 2001. *Algebraic Analysis for Non-identifiable Learning Machines*. Neural Computation, 13, (4) 899–933.