Hybrid Online Inference with Adaptor Grammars

Ke Zhai[1], **Jordan Boyd-Graber**[2], **Shay B. Cohen**[3]

[1]Computer Science
University of Maryland
College Park, MD USA
zhaike@cs.umd.edu

[2]Computer Science
University of Colorado
Boulder, CO USA
jordan.boyd.graber@colorado.edu

[3]School of Informatics
University of Edinburgh
Edinburgh, Scotland, UK
scohen@inf.ed.ac.uk

Abstract

Adaptor grammars are a flexible, powerful formalism for defining nonparametric, unsupervised models of grammar productions. This flexibility comes at the cost of expensive inference. We address the difficulty of inference through an online algorithm which uses a hybrid of Markov chain Monte Carlo and variational inference. We show that this inference strategy is more scalable than past approaches.

1 Introduction

Nonparametric Bayesian models are effective tools to discover latent structure in data [1]. In this work, we focus on adaptor grammars [2], which is a syntactic nonparametric models based on PCFG. They provide a flexible and useful tool in understanding unstructured data or data where the structure is latent, like language. They have been successfully applied for topic modeling [3], discovering perspective [4], segmentation [5], and grammar induction [6]. Despite its modeling advantage, the inference is slow and often not scalable to large datasets. A common approach to address this computational bottleneck is through variational inference [7]. One of the advantages of variational inference is that it can be easily parallelized [8] or transformed into an online algorithm [9], which often converges in fewer iterations than batch variational inference.

Past variational inference techniques for adaptor grammars assume a preprocessing step that looks at *all available data* to establish the support of these nonparametric distributions [6]. Thus, these past approaches are not directly amenable to online inference. Markov chain Monte Carlo (MCMC) inference, an alternative to variational inference, does not have this disadvantage. We apply stochastic hybrid inference [10] to adaptor grammars by interleaving MCMC inference *inside* variational inference. This preserves the scalability of variational inference while adding the sparse statistics and improved exploration MCMC provides.

Our inference algorithm for adaptor grammars starts with a variational algorithm similar to [6] and adds hybrid sampling within variational inference. We further extend it to online setting and process examples in small batches from a data stream. The algorithm dynamically extends the underlying approximate posterior distributions as more data are observed. This makes the algorithm flexible, scalable, and amenable to datasets that cannot be examined exhaustively because of their size. Please see [11] for a detailed version.

2 Adaptor Grammars

A *Pitman-Yor Adaptor grammar* (PYAG) forms the adapted tree distributions H_c using a *Pitman-Yor process* [12, PY]. A draw $H_c \equiv (\boldsymbol{\pi}_c, \boldsymbol{z}_c)$ is formed by the stick breaking process [13] parametrized by scale parameter a, discount factor b, and base distribution G_c:

$$\pi'_k \sim \text{Beta}(1-b, a+kb), \qquad z_k \sim G_c, \qquad \pi_k \equiv \pi'_k \textstyle\prod_{j=1}^{k-1}(1-\pi'_j), \qquad H \equiv \textstyle\sum_k \pi_k \delta_{z_k}. \tag{1}$$

Intuitively, the distribution H_c, which is often referred to as *grammaton*, is a discrete reconstruction of the atoms sampled from G_c—hence, reweights G_c. Grammaton H_c assigns non-zero stick-breaking weights $\boldsymbol{\pi}$ to a countably infinite number of parse trees $\boldsymbol{z}$. We describe learning these grammatons in Section 3.

More formally, a PYAG is a quintuple $\boldsymbol{\mathcal{A}} = \langle \boldsymbol{\mathcal{G}}, \boldsymbol{M}, \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{\alpha} \rangle$ with: a PCFG $\boldsymbol{\mathcal{G}}$; a set of adapted nonterminals $\boldsymbol{M} \subseteq \boldsymbol{N}$; Pitman-Yor process parameters a_c, b_c at each adaptor $c \in \boldsymbol{M}$ and Dirichlet parameters $\boldsymbol{\alpha}_c$ for each nonterminal $c \in \boldsymbol{N}$. We also assume an order on the adapted nonterminals, $c_1, \dots, c_{|\boldsymbol{M}|}$ such that c_j is not reachable from c_i in a derivation if $j > i$.

Given a PYAG $\boldsymbol{\mathcal{A}}$, the joint probability for a set of sentences $\boldsymbol{X}$ and its collection of trees $\boldsymbol{T}$ is

$$p(\boldsymbol{X}, \boldsymbol{T}, \boldsymbol{\pi}, \boldsymbol{\theta}, \boldsymbol{z} | \boldsymbol{\mathcal{A}}) = \prod_{c \in \boldsymbol{M}} p(\boldsymbol{\pi}_c | a_c, b_c) p(\boldsymbol{z}_c | G_c) \prod_{c \in \boldsymbol{N}} p(\boldsymbol{\theta}_c | \alpha_c) \prod_{x_d \in \boldsymbol{X}} p(x_d, t_d | \boldsymbol{\theta}, \boldsymbol{\pi}, \boldsymbol{z}),$$

where x_d and t_d represent the d^{th} observed string and its corresponding parse.

3 Online Hybrid Variational-MCMC Inference

Variational inference posits a variational distribution over the latent variables in the model; this in turn induces an "evidence lower bound" (ELBO, $\mathcal{L}$) as a function of a variational distribution q, a lower bound on the marginal log-likelihood. Variational inference optimizes this objective function with respect to the parameters that define q.

In this section, we derive coordinate-ascent updates for these variational parameters. We strategically use MCMC sampling to compute the expectation of q *over parse trees* z. This produces a sparse approximation of the variational distribution, which improves both scalability and performance. Sparse distributions are easier to store and transmit in implementations, which improves scalability. [10] also show that sparse representations improve performance.

Variational Lower Bound We posit a mean-field variational distribution:

$$q(\boldsymbol{\pi}, \boldsymbol{\theta}, \boldsymbol{T} | \gamma, \boldsymbol{\nu}, \boldsymbol{\phi}) = \prod_{c \in \boldsymbol{M}} \prod_{i=1}^{\infty} q(\pi'_{c,i} | \nu^1_{c,i}, \nu^2_{c,i}) \prod_{c \in \boldsymbol{N}} q(\boldsymbol{\theta}_c | \gamma_c) \prod_{x_d \in \boldsymbol{X}} q(t_d | \boldsymbol{\phi}_d), \tag{2}$$

where $\pi'_{c,i}$ is drawn from a variational Beta distribution parameterized by $\nu^1_{c,i}, \nu^2_{c,i}$; and $\boldsymbol{\theta}_c$ is from a variational Dirichlet prior $\boldsymbol{\gamma}_c \in \mathbb{R}_+^{|\boldsymbol{R}(c)|}$. Index i ranges over a possibly infinite number of adapted rules. The parse for the d^{th} observation, t_d is modeled by a multinomial $\boldsymbol{\phi}_d$, where $\phi_{d,i}$ is the probability generating the i^{th} phrase-structure tree $t_{d,i}$.

The variational distribution over latent variables induces the following ELBO on the likelihood:

$$\begin{aligned} \mathcal{L}(\boldsymbol{z}, \boldsymbol{\pi}, \boldsymbol{\theta}, \boldsymbol{T}, \boldsymbol{D}; \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{\alpha}) &= \textstyle\sum_{c \in \boldsymbol{N}} \mathbb{E}_q[\log p(\boldsymbol{\theta}_c | \boldsymbol{\alpha}_c)] + \sum_{c \in \boldsymbol{M}} \sum_{i=1}^{\infty} \mathbb{E}_q[\log p(\pi'_{c,i} | a_c, b_c)] \\ &+ \textstyle\sum_{c \in \boldsymbol{M}} \sum_{i=1}^{\infty} \mathbb{E}_q[\log p(z_{c,i} \,|\, \boldsymbol{\pi}, \boldsymbol{\theta})] + \sum_{x_d \in \boldsymbol{X}} \mathbb{E}_q[\log p(x_d, t_d \,|\, \boldsymbol{\pi}, \boldsymbol{\theta}, \boldsymbol{z})] + H[q(\boldsymbol{\theta}, \boldsymbol{\pi}, \boldsymbol{T})] \end{aligned} \tag{3}$$

where $H[\bullet]$ is the entropy function. To make this lower bound tractable, we truncate the distribution over π to a finite set [14] for each adapted nonterminal $c \in \boldsymbol{M}$, i.e., $\pi'_{c,K_c} \equiv 1$ for some index K_c. Each weight $\pi_{c,i}$ is associated with an atom $z_{c,i}$, a subtree rooted at c. We call the ordered set of $z_{c,i}$ the *truncated nonterminal grammaton* (TNG). Each adapted nonterminal $c \in \boldsymbol{M}$ has its own TNG_c. The i^{th} subtree in TNG_c is denoted $\text{TNG}_c(i)$.

Stochastic MCMC Inference Each observation x_d has an associated variational multinomial distribution $\boldsymbol{\phi}_d$ over trees $\boldsymbol{t}_d$ that can yield observation x_d with probability $\phi_{d,i}$. Holding all other variational parameters fixed, the coordinate-ascent update [10, 15] for $\phi_{d,i}$ is

$$\phi_{d,i} \propto \exp\{\mathbb{E}_q^{\neg \boldsymbol{\phi}_d}[\log p(t_{d,i} | x_d, \boldsymbol{\pi}, \boldsymbol{\theta}, \boldsymbol{z})]\}, \tag{4}$$

where $\phi_{d,i}$ is the probability generating the i^{th} phrase-structure tree $t_{d,i}$ and $\mathbb{E}_q^{\neg \boldsymbol{\phi}_d}[\bullet]$ is the expectation with respect to the variational distribution q, excluding the value of $\boldsymbol{\phi}_d$.

We apply stochastic variational inference [10, 16] to sample from this distribution. This produces a set of sampled trees $\boldsymbol{\sigma}_d \equiv \{\sigma_{d,1}, \dots, \sigma_{d,k}\}$. From this set of trees we can approximate our variational distribution over trees $\boldsymbol{\phi}$ using the empirical distribution $\boldsymbol{\sigma}_d$, i.e.,

$$\phi_{d,i} \propto \mathbb{I}[\sigma_{d,j} = t_{d,i}, \forall \sigma_{d,j} \in \boldsymbol{\sigma}_d]. \tag{5}$$

This leads to a sparse approximation of variational distribution $\boldsymbol{\phi}$. We use 10 samples in experiments.

Previous inference strategies [2, 17] for adaptor grammars have used sampling. Sampling requires a derived PCFG $\boldsymbol{\mathcal{G}'}$ that approximates the distribution over tree derivations conditioned on a yield. It includes the original PCFG rules $\boldsymbol{R} = \{c \to \beta\}$ that define the base distribution and the new adapted productions $\boldsymbol{R'} = \{c \Rightarrow z, z \in \text{TNG}_c\}$. Under $\boldsymbol{\mathcal{G}'}$, the probability θ' of adapted production $c \Rightarrow z$ is

$$\log \theta'_{c\Rightarrow z} = \begin{cases} \mathbb{E}_q[\log \pi_{c,i}], & \text{if } \text{TNG}_c(i) = z \\ \mathbb{E}_q[\log \pi_{c,K_c}] + \mathbb{E}_q[\log \theta_{c\Rightarrow z}], & \text{otherwise} \end{cases} \tag{6}$$

where K_c is the truncation level of TNG_c and π_{c,K_c} represents the left-over stick weights in the stick-breaking process for adaptor $c \in \boldsymbol{M}$. $\theta_{c\Rightarrow z}$ represents the probability of generating tree $c \Rightarrow z$ under the base distribution. See also [18].

The expectation of $\pi_{c,i}$ under the truncated variational stick-breaking distribution is

$$\mathbb{E}_q[\log \pi_{a,i}] = \Psi(\nu^1_{a,i}) - \Psi(\nu^1_{a,i} + \nu^2_{a,i}) + \textstyle\sum_{j=1}^{i-1}(\Psi(\nu^2_{a,j}) - \Psi(\nu^1_{a,j} + \nu^2_{a,j})), \tag{7}$$

and the expectation of generating the phrase-structure tree $a \Rightarrow z$ based on PCFG productions is

$$\mathbb{E}_q[\log \theta_{a\Rightarrow z}] = \textstyle\sum_{c\to\beta\in a\Rightarrow z} \left(\Psi(\gamma_{c\to\beta}) - \Psi(\sum_{c\to\beta'\in \boldsymbol{R}_c} \gamma_{c\to\beta'})\right) \tag{8}$$

where $\Psi(\bullet)$ is the digamma function, and $c \to \beta \in a \Rightarrow z$ represents all PCFG productions in the phrase-structure tree $a \Rightarrow z$. This PCFG can compose arbitrary subtrees and thus discover new trees that better describe the data, even if those trees are not part of the TNG. This is equivalent to creating a "new table" in MCMC inference and provides *truncation-free* variational updates [19] by sampling a unseen subtree with adapted nonterminal $c \in \boldsymbol{M}$ at the root. This frees our model from preprocessing to initialize truncated grammatons in [6]. This stochastic approach has the advantage of creating sparse distributions [19]. In addition, it also preserves the independent structure from varaitional distributions and can be easily parallelized.

Calculating Expected Rule Counts Let us refer the multiset of all adapted productions as $M(t_{d,i})$ and the multiset of PCFG productions as $N(t_{d,i})$. For every observation x_d, we compute:

1: the expected number of productions within the TNG of adapted production $a \Rightarrow z_{a,i}$:

$$f_d(a \Rightarrow z_{a,i}) = \textstyle\sum_k \big(\phi_{d,k} \underbrace{|a \Rightarrow z_{a,i} : a \Rightarrow z_{a,i} \in M(t_{d,k})|}_{\text{count of rule } a \Rightarrow z_{a,i} \text{ in tree } t_{d,k}}\big).$$

2: the expected counts of PCFG productions $\boldsymbol{R}$ that defines the *base* distribution:

$$g_d(a \to \beta) = \textstyle\sum_k \left(\phi_{d,k} \left|a \to \beta : a \to \beta \in N(t_{d,k})\right|\right).$$

3: a set of productions that are newly discovered by the sampler and not in the TNG:

$$h_d(c \Rightarrow z_{c,i}) = \textstyle\sum_k \left(\phi_{d,k} \left|c \Rightarrow z_{c,i} : c \Rightarrow z_{c,i} \notin M(t_{d,k})\right|\right).$$

These counts can be computed by aggregating over different machines in distributed environment.

Variational Updates Given the sampled sparse vectors $\boldsymbol{\phi}$, we update all variational parameters as

$$\begin{aligned} \gamma_{a\to\beta} =& \alpha_{a\to\beta} + \textstyle\sum_{x_d\in \boldsymbol{X}} g_d(a \to \beta) + \sum_{b\in \boldsymbol{M}} \sum_{i=1}^{K_b} n(a \to \beta, z_{b,i}), \\ \nu^1_{a,i} =& 1 - b_a + \textstyle\sum_{x_d\in \boldsymbol{X}} f_d(a \Rightarrow z_{a,i}) + \sum_{b\in \boldsymbol{M}} \sum_{k=1}^{K_b} n(a \Rightarrow z_{a,i}, z_{b,k}), \\ \nu^2_{a,i} =& a_a + ib_a + \textstyle\sum_{x_d\in \boldsymbol{X}} \sum_{j=1}^{K_a} f_d(a \Rightarrow z_{a,j}) + \sum_{b\in \boldsymbol{M}} \sum_{k=1}^{K_b} \sum_{j=1}^{K_a} n(a \Rightarrow z_{a,j}, z_{b,k}), \end{aligned}$$

where $n(r, t)$ is the expected number of times production r is in tree t, estimated during sampling. We update our PCFG hyperparameter $\boldsymbol{\alpha}$, PYGEM hyperparameters $\boldsymbol{a}$ and $\boldsymbol{b}$ as in [6].

Online Variational Inference In online case, we assume data arrive in minibatches $\boldsymbol{B}$ (a set of sentences). Hence, we accumulate expected counts

$$\tilde{f}^{(l)}(a \Rightarrow z_{a,i}) = (1 - \epsilon) \cdot \tilde{f}^{(l-1)}(a \Rightarrow z_{a,i}) + \epsilon \cdot \tfrac{|\boldsymbol{X}|}{|\boldsymbol{B}_l|} \textstyle\sum_{x_d\in \boldsymbol{B}_l} f_d(a \Rightarrow z_{a,i}), \tag{9}$$

$$\tilde{g}^{(l)}(a \to \beta) = (1 - \epsilon) \cdot \tilde{g}^{(l-1)}(a \to \beta) + \epsilon \cdot \tfrac{|\boldsymbol{X}|}{|\boldsymbol{B}_l|} \textstyle\sum_{x_d\in \boldsymbol{B}_l} g_d(a \to \beta), \tag{10}$$

with *decay factor* $\epsilon \in (0, 1)$ to guarantee convergence. We set it to $\epsilon = (\tau + l)^{-\kappa}$, where l is the minibatch counter. The *decay inertia* τ prevents premature convergence, and *decay rate* κ controls the speed of change in sufficient statistics [9]. We recover batch variational approach when $\boldsymbol{B} = \boldsymbol{D}$ and $\kappa = 0$. The variables $\tilde{f}^{(l)}$ and $\tilde{g}^{(l)}$ are accumulated sufficient statistics of adapted and unadapted productions after processing minibatch $\boldsymbol{B}_l$. The updates for variational parameters become

$$\begin{aligned}
\gamma_{a\to\beta} =& \alpha_{a\to\beta} + \tilde{g}^{(l)}(a \to \beta) + \textstyle\sum_{b\in\boldsymbol{M}} \sum_{i=1}^{K_b} n(a \to \beta, z_{b,i}), && (11)\\
\nu^1_{a,i} =& 1 - b_a + \tilde{f}^{(l)}(a \Rightarrow z_{a,i}) + \textstyle\sum_{b\in\boldsymbol{M}} \sum_{k=1}^{K_b} n(a \Rightarrow z_{a,i}, z_{b,k}), && (12)\\
\nu^2_{a,i} =& a_a + ib_a + \textstyle\sum_{j=1}^{K_a} \tilde{f}^{(l)}(a \Rightarrow z_{a,j}) + \sum_{b\in\boldsymbol{M}} \sum_{k=1}^{K_b} \sum_{j=1}^{K_a} n(a \Rightarrow z_{a,j}, z_{b,k}), && (13)
\end{aligned}$$

where K_a is the size of the TNG at adaptor $a \in \boldsymbol{M}$.

Refining the Truncation Our model does not require a preprocessing step to initialize the TNGs, rather, it constructs and expands all TNGs on the fly. To prevent the TNG from growing unwieldy, we prune TNG after every u minibatches. As a result, we need to impose an ordering over all the parse trees in the TNG. Similar to [20], we impose a reward term for longer phrases in addition to $\tilde{f}$ and sort all adapted productions in TNG$_a$ using the ranking score

$$\Lambda(a \Rightarrow z_{a,i}) = \tilde{f}^{(l)}(a \Rightarrow z_{a,i}) \cdot \log(\epsilon \cdot |s| + 1),$$

where $|s|$ is the number of yields in production $a \Rightarrow z_{a,i}$. Because ϵ decreases each minibatch, the reward for long phrases diminishes. This is similar to an annealed version of [6]—where the reward for long phrases is fixed. After sorting, we remove all but the top K_a adapted productions.

4 Experiments

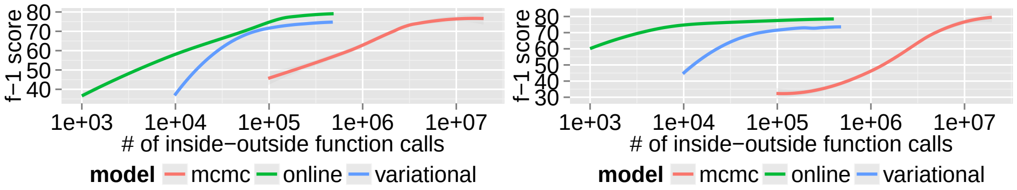

Figure 1: Word segmentation accuracy measured by word token F_1 scores on *brent* corpus of three approaches against number of inside-outside function call using *unigram* (left) and *collocation* (right) grammars in [5]. Our ONLINE settings are batch size $B = 20$, decay inertia $\tau = 128$, decay rate $\kappa = 0.6$ for *unigram* grammar; and minibatch size $B = 5$, decay inertia $\tau = 256$, decay rate $\kappa = 0.8$ for *collocation* grammar. TNGs are refined at interval $u = 50$. Truncation size is set to $K_{\text{Word}} = 1.5k$ and $K_{\text{Colloc}} = 3k$. We observe similar behavior under $\kappa = \{0.7, 0.9, 1.0\}$, $\tau = \{32, 64, 512\}$, $B = \{10, 50\}$ and $u = \{10, 20, 100\}$.

We implement our online adaptor grammar model (ONLINE) in Python and compare it against both MCMC [5, MCMC] and the variational inference [6, VARIATIONAL]. We focus on the task of word segmentation, which focuses on identifying word boundaries from a sequence of characters. We evaluate all three models on the standard Brent version of the Bernstein-Ratner corpus [21, 22, *brent*]. The dataset contains $10k$ sentences, $1.3k$ distinct words, and 72 distinct characters. We compare the results on both *unigram* and *collocation* grammars introduced in [5].

Figure 1 illustrates the word segmentation accuracy in terms of word token F_1-scores on *brent* against the number of inside-outside function calls for all three approaches using *unigram* and *collocation* grammars. In both cases, our ONLINE approach converges faster than MCMC and VARIATIONAL approaches, yet yields comparable or better performance when seeing more data. We also evaluate these three approaches on much larger datasets in addition to the *brent* corpus [11].

5 Conclusion

Adaptor grammars offer a flexible and quick way to prototype and test new models. We have presented a new online, hybrid inference scheme for adaptor grammars. We show that it is able to faster discover useful structure in text than past approaches.

Acknowledgments

We would like to thank the anonymous reviewers, Kristina Toutanova, Mark Johnson, and Ke Wu for insightful discussions. This work was supported by NSF Grant CCF-1018625. Boyd-Graber is also supported by NSF Grant IIS-1320538. Any opinions, findings, conclusions, or recommendations expressed here are those of the authors and do not necessarily reflect the view of the sponsor.

References

[1] Müller, P., F. A. Quintana. Nonparametric Bayesian data analysis. *Statistical Science*, 19(1), 2004.

[2] Johnson, M., T. L. Griffiths, S. Goldwater. Adaptor grammars: A framework for specifying compositional nonparametric Bayesian models. In *Proceedings of Advances in Neural Information Processing Systems*. 2006.

[3] Johnson, M. PCFGs, topic models, adaptor grammars and learning topical collocations and the structure of proper names. In *Proceedings of the Association for Computational Linguistics*. 2010.

[4] Hardisty, E., J. Boyd-Graber, P. Resnik. Modeling perspective using adaptor grammars. In *Proceedings of Emperical Methods in Natural Language Processing*. 2010.

[5] Johnson, M., S. Goldwater. Improving nonparameteric Bayesian inference: experiments on unsupervised word segmentation with adaptor grammars. In *Conference of the North American Chapter of the Association for Computational Linguistics*. 2009.

[6] Cohen, S. B., D. M. Blei, N. A. Smith. Variational inference for adaptor grammars. In *Conference of the North American Chapter of the Association for Computational Linguistics*. 2010.

[7] Wainwright, M. J., M. I. Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1–2):1–305, 2008.

[8] Nallapati, R., W. Cohen, J. Lafferty. Parallelized variational EM for latent Dirichlet allocation: An experimental evaluation of speed and scalability. In *ICDMW*. 2007.

[9] Hoffman, M., D. M. Blei, F. Bach. Online learning for latent Dirichlet allocation. In *Proceedings of Advances in Neural Information Processing Systems*. 2010.

[10] Mimno, D., M. Hoffman, D. Blei. Sparse stochastic inference for latent Dirichlet allocation. In *Proceedings of the International Conference of Machine Learning*. 2012.

[11] Zhai, K., J. Boyd-Graber, S. B. Cohen. Online adaptor grammars with hybrid inference. *Transactions of the Association for Computational Linguistics*, 2(0):465–476, 2014.

[12] Pitman, J., M. Yor. The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator. *Annals of Probability*, 25(2):855–900, 1997.

[13] Sudderth, E. B., M. I. Jordan. Shared segmentation of natural scenes using dependent Pitman-Yor processes. In *Proceedings of Advances in Neural Information Processing Systems*. 2008.

[14] Blei, D. M., M. I. Jordan. Variational inference for Dirichlet process mixtures. *Journal of Bayesian Analysis*, 1(1):121–144, 2005.

[15] Bishop, C. M. *Pattern Recognition and Machine Learning*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[16] Hoffman, M., D. M. Blei, C. Wang, et al. Stochastic variational inference. In *Journal of Machine Learning Research*. 2013.

[17] Börschinger, B., M. Johnson. Using rejuvenation to improve particle filtering for bayesian word segmentation. In *Proceedings of the Association for Computational Linguistics*. 2012.

[18] Cohen, S. B. *Computational Learning of Probabilistic Grammars in the Unsupervised Setting*. Ph.D. thesis, Carnegie Mellon University, 2011.

[19] Wang, C., D. M. Blei. Truncation-free online variational inference for Bayesian nonparametric models. In *Proceedings of Advances in Neural Information Processing Systems*. 2012.

[20] Mochihashi, D., T. Yamada, N. Ueda. Bayesian unsupervised word segmentation with nested pitman-yor language modeling. In *Proceedings of the Association for Computational Linguistics*. 2009.

[21] Bernstein-Ratner, N. The phonology of parent child speech. *Children's language*, 6:159–174, 1987.

[22] Brent, M. R., T. A. Cartwright. Distributional regularity and phonotactic constraints are useful for segmentation. vol. 61, pages 93–125. 1996.