Online Variational Inference for the Hierarchical Dirichlet Process

Presented by Chong Wang
Joint work with John Paisley and David M. Blei

Computer Science Department, Princeton University

AISTATS 2011, Ft. Lauderdale, FL, USA

Why Online Variational Inference for the HDP

1. Hierarchical Dirichlet Process (HDP) (Teh et al. [2007a]),
   - can be used as a powerful mixed-membership model for data like documents and images.
   - can infer the number of components using posterior inference.
2. Posterior inference for the HDP model is intractable.
   - 1) Gibbs sampling and 2) variational inference.
   - These are *batch* algorithms, requiring multiple passes through the data — limited for massive scale applications and streaming data.

Online variational inference lets us (approximately) infer the posterior for massive and streaming data.

Outline

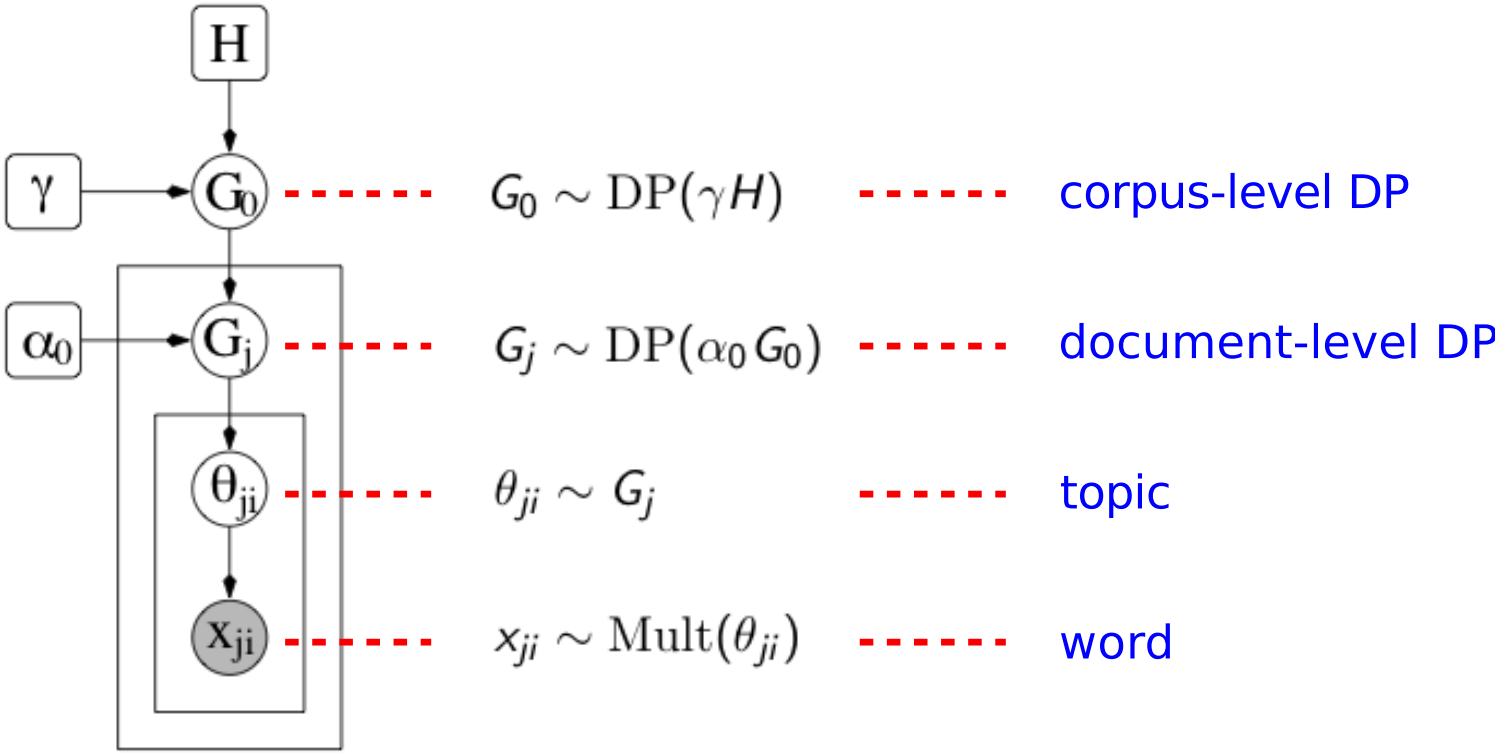
Why a new batch variational inference algorithm?

- Existing batch variational inference algorithms require complicated approximations or numerical optimization.(Teh et al. [2007b], Liang et al. [2007], Boyd-Graber and Blei [2009])
- We need a new variational inference algorithm that allows *simple and closed-form* updates for variational parameters.
- Then we use stochastic natural gradient similar to online variational LDA.(Hoffman et al. [2010])

Next, we review the HDP for document modeling — the HDP topic model, and then describe new algorithm.

The HDP topic model

We focus on a topic model with a two-level HDP.



This representation, however, is *implicit* and hard to work with for variational inference. So we use stick-breaking construction.

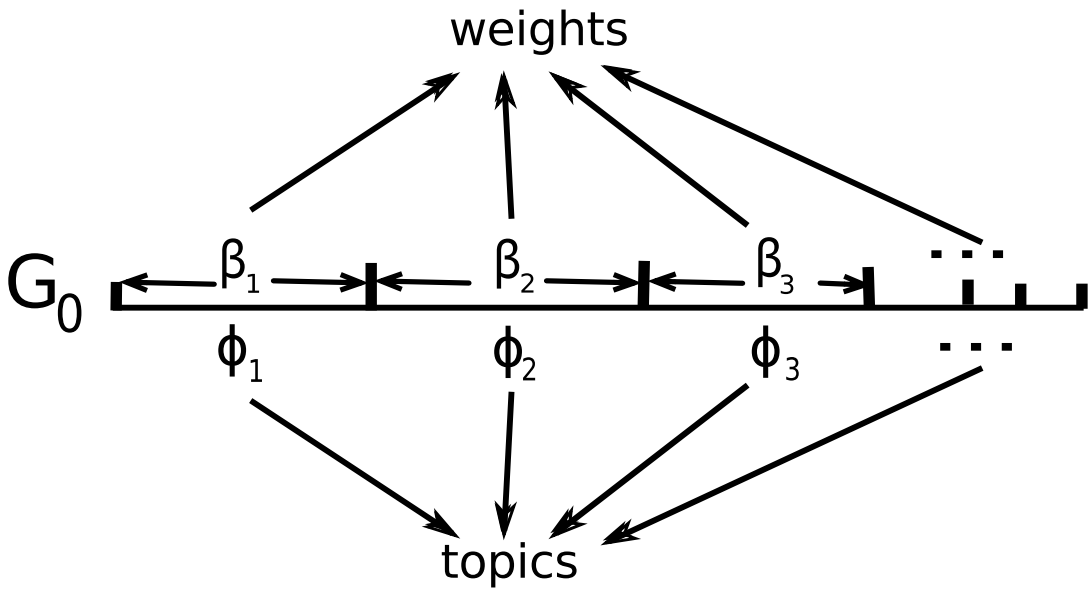
Sethuraman's stick breaking for the DP

This is an explicit representation. For the corpus-level DP, $G_0 \sim \mathrm{DP}(\gamma H)$, Sethuraman's stick breaking construction gives,

$\phi_k \sim H$, —— atoms/topics

$\boldsymbol{\beta} \sim \mathrm{GEM}(\gamma)$ —— $\beta'_k \sim \mathrm{Beta}(1, \gamma)$, $\beta_k = \beta'_k \prod_{l=1}^{k-1}(1 - \beta'_l)$

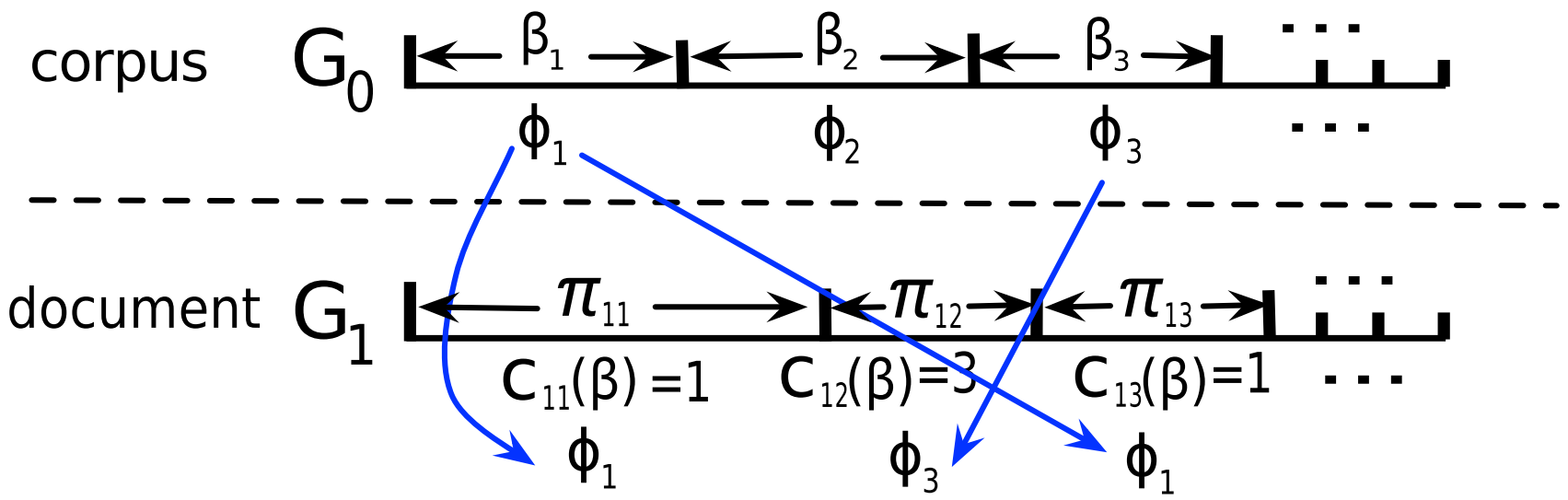
$G_0 = \sum_{k=1}^{\infty} \beta_k \delta_{\phi_k}$.



Draw a sample G_0: $\psi \sim G_0 <=> c \sim \mathrm{Mult}(\boldsymbol{\beta})$ and $\psi = \phi_c$.

Sethuraman's stick breaking for the HDP

For the HDP, we further apply Sethuraman's stick breaking for document j, since $G_j \sim \mathrm{DP}(\alpha_0 G_0)$ (Fox et al. [2008]),



What's important:

- corpus-level sticks $\boldsymbol{\beta} \sim \mathrm{GEM}(\gamma)$.
- document-level sticks $\boldsymbol{\pi}_j \sim \mathrm{GEM}(\alpha_0)$.

These are *decoupled*, unlike the representation in (Teh et al. [2007b]).

A new batch variational inference algorithm

hidden variables

(Φ, β') ----- G_0 ----- corpus-level DP

(π', c) ----- G_j ----- document-level DP

(z) ---------- word-level topic

1. Corpus level:
   - topic distribution — ϕ,
   - stick proportions — β'.
2. Document level:
   - stick proportions — $\pi' = (\pi'_j)_{j=1}^M$,
   - topic indicators — $\mathbf{c} = (\mathbf{c}_j)_{j=1}^M$,
3. Word level:
   - topic indexes for words — $\mathbf{z}$.

A new batch variational inference algorithm – cont'

Fully factorized variational distribution to approximate the full posterior

$$q(\beta', \phi, \pi', \mathbf{c}, \mathbf{z}) = \boxed{q(\beta')q(\phi)} \times \boxed{q(\pi')q(\mathbf{c})} \times \boxed{q(\mathbf{z})}$$

$q(\beta')q(\phi)$ — *corpus* level variational distribution

$q(\pi')q(\mathbf{c})$ — *document* level variational distribution

$q(\mathbf{z})$ — *word* level variational distribution

Find $\arg\min_q KL(q||p)$ — p is the true posterior

1. An coordinate ascent algorithm optimizes the lower bound of marginal likelihood of the observed data.
2. Truncations apply to both corpus-level (K) and document-level (T).
3. *All parameter updates are in closed form due to full conjugacy*.

The batch variational inference algorithm — flow

1: **while** stopping criterion is not met **do**
2: Reset the sufficient statistic.
3: **for** each document **do**
4: Update *document-level* and its *word-level* variational parameters.
5: Update sufficient statistic.
6: **end for**
7: Update *corpus-level* parameters using the sufficient statistic from *all the documents*.
8: **end while**

- Multiple passes of the entire data — limited for large scale applications or streaming data.
- Our solution: stochastic (natural) gradient — the closed-form batch algorithm leads to very simple online updates.

The general online variational inference — flow

1: **while** stopping criterion is not met **do**
2: Fetch a random document.
3: Update *document-level* and its *word-level* variational parameters.
4: Update *corpus-level* parameters using given the statistic *only from this document.*
5: **end while**

- Updating document *and word-level* parameters does not change.
- Updating the corpus-level parameters using the stochastic (natural) gradient.

Stochastic gradient

- Stochastic gradient, one type of stochastic optimization methods (Robbins and Monro [1951]) , is especially efficient for large scale applications.
- Stochastic gradient proceeds by iteratively taking a random subset of the data, and updating the model parameters w.r.t. to this subset.
- The gradient that we are following is a noisy estimate using only the subset of the data.
- A decaying learning rate is required to guarantee convergence.
- We apply stochastic gradient to the variational objective of the HDP.

Stochastic gradient — cont'

Let A_t be a positive definite matrix and ρ_t be the learning rate.

- In batch variational inference, we optimize a lower bound $\mathcal{L}$, if we do gradient ascent,

$$\partial\mathcal{L}/\partial\lambda \text{ — gradient}$$

$$\lambda_{t+1} \leftarrow \lambda_t + \rho_t A_t \boxed{\partial\mathcal{L}/\partial\lambda} \text{ — gradient ascent}$$

- For online variational inference, we first notice

$$\mathcal{L} = \sum_j \mathcal{L}_j = \frac{1}{D}\sum_{j=1}^{D} D\mathcal{L}_j$$

Stochastic gradient:

$$\partial\mathcal{L}/\partial\lambda \approx \boxed{\partial D\mathcal{L}_j/\partial\lambda} \text{ — a noisy estimate from a single doc}$$

If A_t is the inverse of Riemannian metric (Amari [1998], Sato [2005]), we obtain the stochastic *natural gradient*.

The online variational inference with natural gradient

We take the topic distribution parameter λ for an example.
Let s_j be the statistic from document j.

- Batch inference through coordinate ascent, by setting $\frac{\partial \mathcal{L}}{\partial \lambda} = 0$,

$$\lambda \leftarrow \boxed{\eta + \sum_{j=1}^{D} s_j}.$$

- Online inference with *natural gradient* — enabled by the new batch inference algorithm, has a similar form,

$$\lambda_{t+1} \leftarrow (1 - \rho_t)\lambda_t + \rho_t \boxed{(\eta + D s_j)},$$

Mini-batches: using multiple samples each time.

The final online variational inference algorithm

1: **Input**: data, learning rate ρ_t and mini-batch size S.
2: **while** stopping criterion is not met **do**
3: Fetch S random documents.
4: Update the variational parameters for every document in this set.
5: Update *corpus-level* parameters using given the stochastic natural gradient. For example,

$$\lambda_{t+1} \leftarrow (1-\rho_t)\lambda_t + \rho_t(\eta + D/S \sum_{j \in S} s_j),$$

6: **end while**

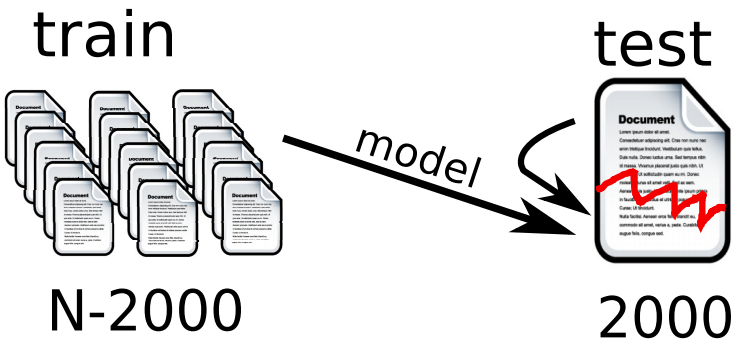
If $\rho_t \equiv 1$ and $S = D$, this is just the batch variational inference.

Data, Metric and Comparisons

Data:

corpus	# documents	# tokens	# vocabulary	year range
Nature	352,549	58 million	4,253	1869-2008
PNAS	82,519	46 million	6,500	1914-2004

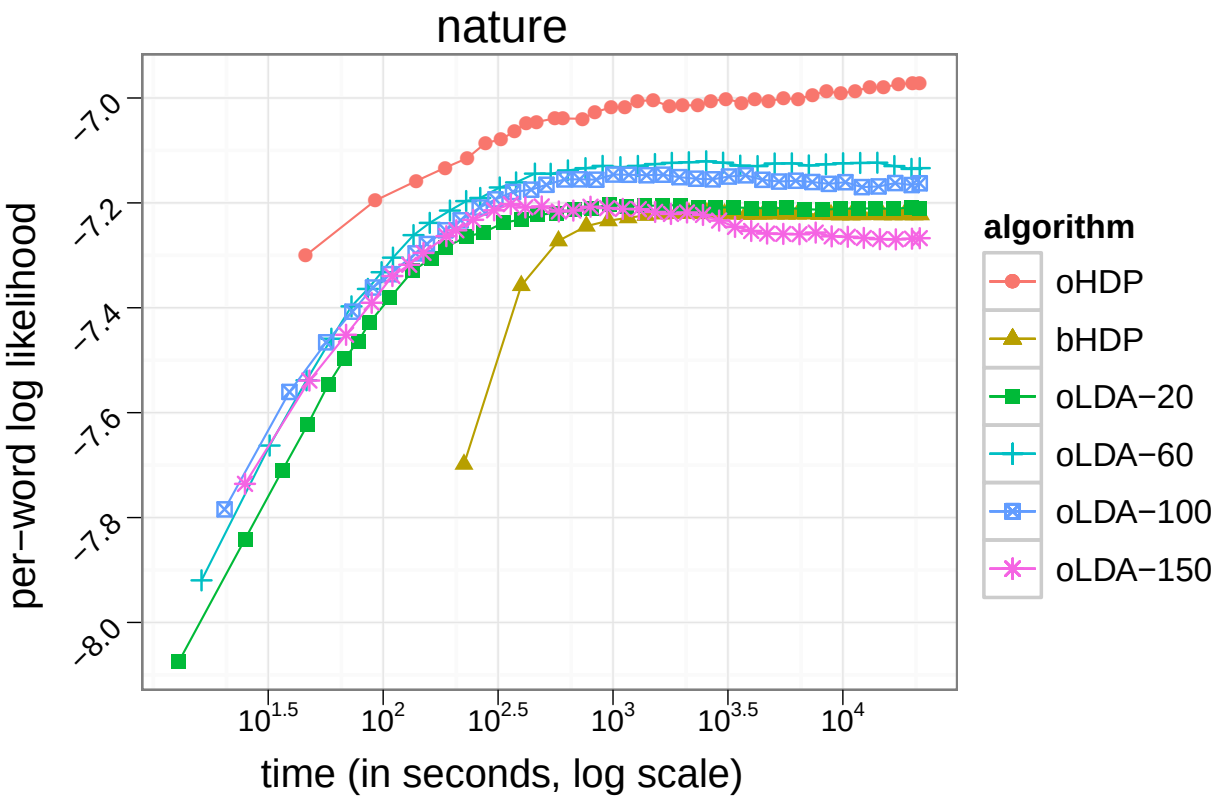
Metric: predictive log likelihood, similar to Asuncion et al. [2009].



Comparisons: compare with online variational LDA (Hoffman et al. [2010]) and batch variational HDP running for 6 hours.

Results on Nature

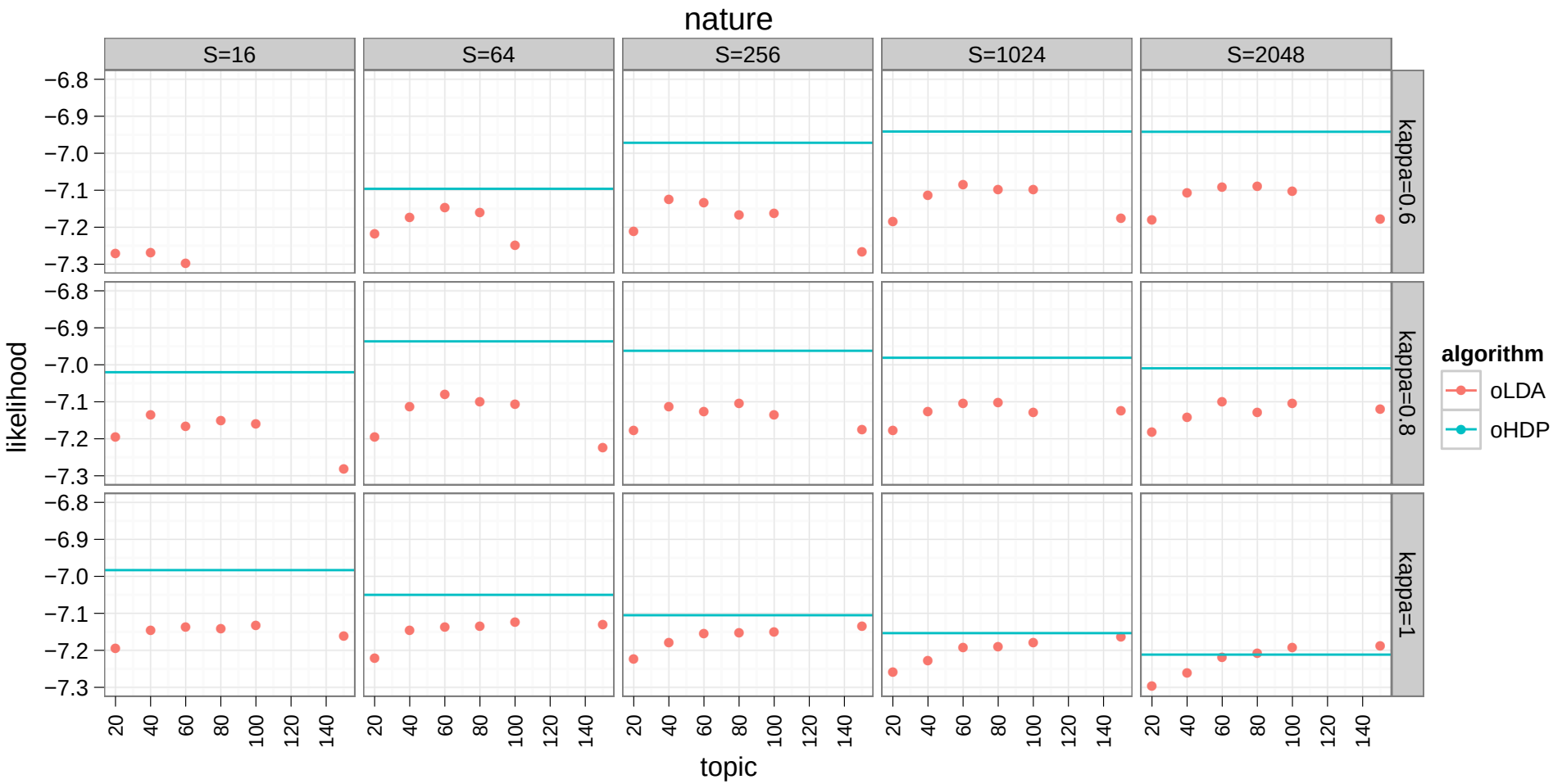
Corpus-level truncation $K = 150$, document-level truncation $T = 15$.
Batch HDP is only trained on a subset since it's too slow.



Online variational HDP uses about 110 topics while online variational LDA uses all of them.

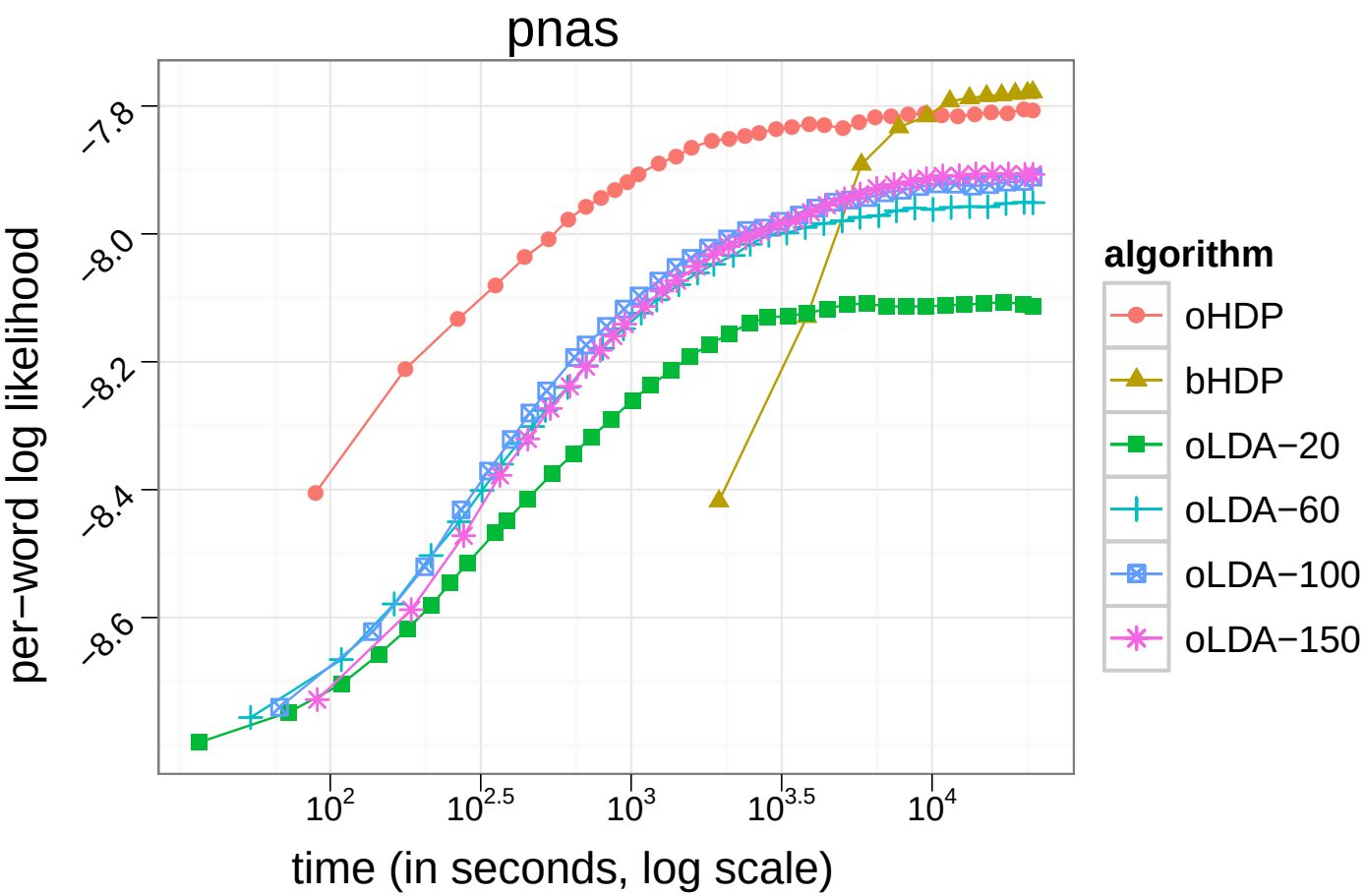
Results on Nature — cont'

We have also tested different learning rates and mini-batch sizes,



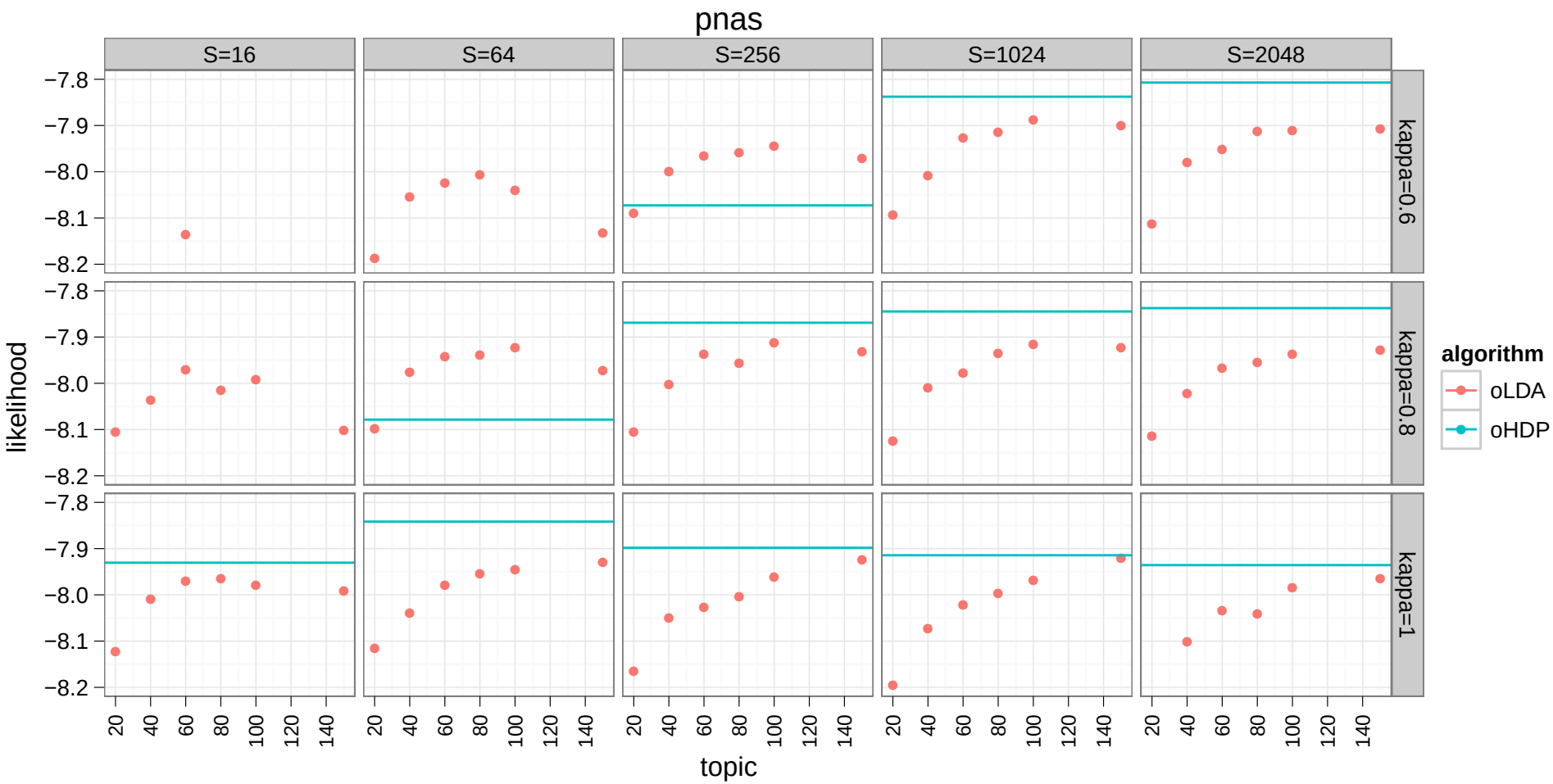
Results on PNAS

Settings are similar to the Nature experiment. Exception: batch HDP is trained on the *whole set*.

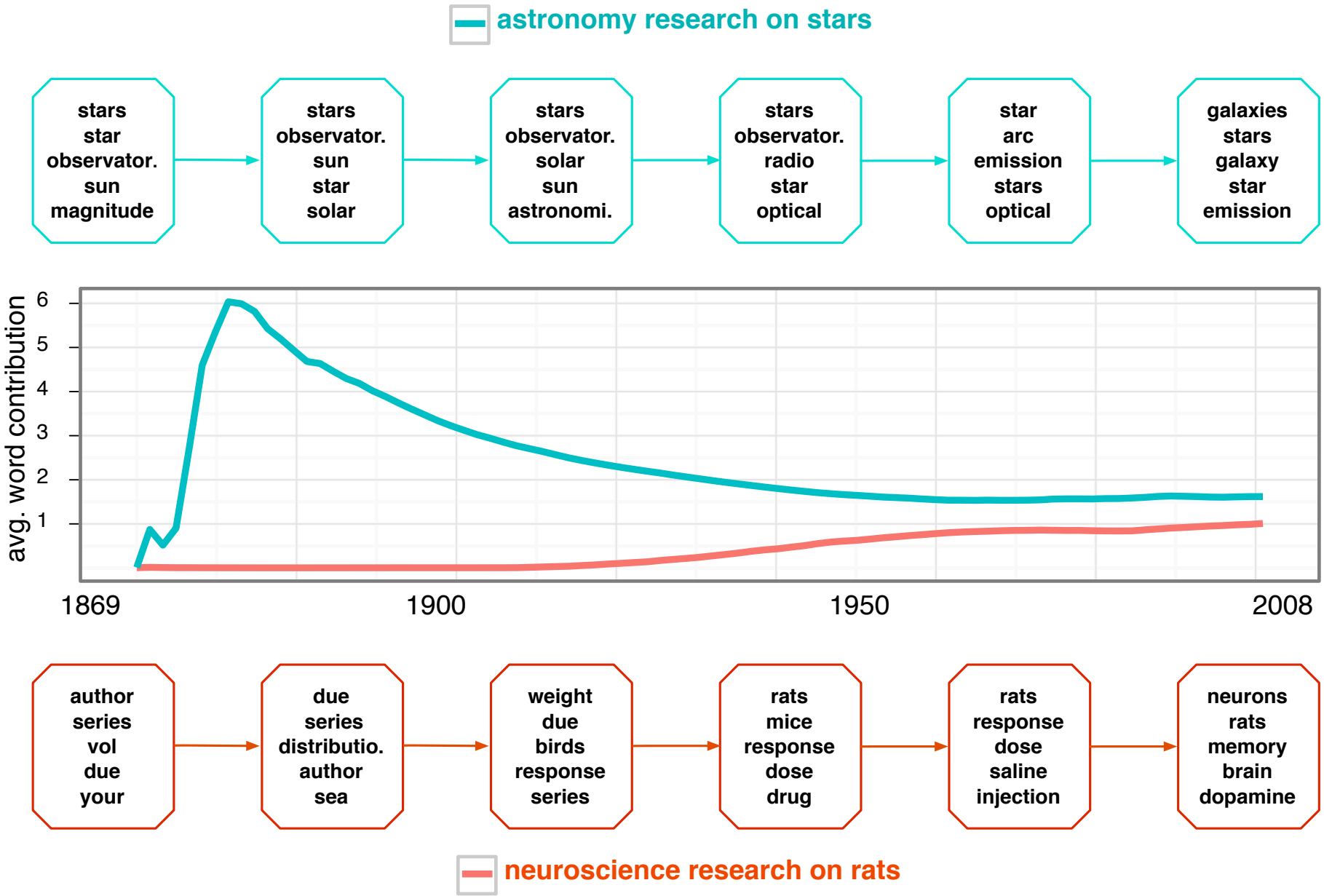


Results on PNAS — cont'

We have also tested different learning rates and mini-batch sizes,



Results on streaming data — simulated on Nature



Summary

We have

1. described a new (batch) variational inference algorithm for the HDP.
2. presented a new online variational inference algorithm for the HDP given the batch algorithm.
3. empirically demonstrated its performance on large-scale applications.

We want to

1. generalize the idea to other Bayesian nonparametric models.
2. automatically grow the truncations on the fly.

References

S.I. Amari. Natural gradient works efficiently in learning. *Neural computation*, 10(2): 251–276, 1998.

A. Asuncion, M. Welling, P. Smyth, and Y. Teh. On smoothing and inference for topic models. In *Uncertainty in Artificial Intelligence*, 2009.

J. Boyd-Graber and D. Blei. Syntactic topic models. In *Neural Information Processing Systems*, 2009.

E. Fox, E. Sudderth, M. Jordan, and A. Willsky. An HDP-HMM for systems with state persistence. In *International Conference on Machine Learning*, 2008.

Matthew Hoffman, David Blei, and Francis Bach. Online inference for latent Drichlet allocation. In *NIPS*, 2010.

P. Liang, S. Petrov, D. Klein, and M. Jordan. The infinite PCFG using hierarchical Dirichlet processes. In *Empirical Methods in Natural Language Processing*, 2007.

Herbert Robbins and Sutton Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):pp. 400–407, 1951.

M.A. Sato. Online model selection based on the variational Bayes. *Neural Computation*, 13(7):1649–1681, 2005.

Y. Teh, M. Jordan, M. Beal, and D. Blei. Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581, 2007a.

Y. Teh, K. Kurihara, and M. Welling. Collapsed variational inference for HDP. In *Neural Information Processing Systems*, 2007b.

There is still something left!

Variational distribution

We use a fully factorized variational distribution and use mean-field variational inference.

$$q(\boldsymbol{\beta}, \boldsymbol{\phi}', \boldsymbol{\pi}', \mathbf{c}, \mathbf{z}) = q(\boldsymbol{\beta}')q(\boldsymbol{\phi})q(\boldsymbol{\pi}')q(\mathbf{c})q(\mathbf{z})$$

This further factorizes into

$$\begin{aligned}
q(\mathbf{c}) &= \textstyle\prod_j \prod_t q(c_{jt}|\varphi_{jt}), \\
q(\mathbf{z}) &= \textstyle\prod_j \prod_n q(z_{jn}|\zeta_{jn}), \\
q(\boldsymbol{\phi}) &= \textstyle\prod_k q(\phi_k|\lambda_k), \\
q(\boldsymbol{\beta}') &= \textstyle\prod_{k=1}^{K-1} q(\beta'_k|u_k, v_k), \\
q(\boldsymbol{\pi}') &= \textstyle\prod_j \prod_{t=1}^{T-1} q(\pi'_{jt}|a_{jt}, b_{jt}),
\end{aligned}$$

Variational updates

All updates are in closed-form.

1. Document-level Updates:

$$
\begin{aligned}
a_{jt} &= 1 + \textstyle\sum_n \zeta_{jnt}, \\
b_{jt} &= \alpha_0 + \textstyle\sum_n \sum_{s=t+1}^{T} \zeta_{jns}, \\
\varphi_{jtk} &\propto \exp\left(\textstyle\sum_n \zeta_{jnt} \mathbb{E}_q\left[\log p(w_{jn}|\phi_k)\right] + \mathbb{E}_q\left[\log \beta_k\right]\right), \\
\zeta_{jnt} &\propto \exp\left(\textstyle\sum_{k=1}^{K} \varphi_{jtk} \mathbb{E}_q\left[\log p(w_{jn}|\phi_k)\right] + \mathbb{E}_q\left[\log \pi_{jt}\right]\right).
\end{aligned}
$$

2. Corpus-level Updates:

$$
\begin{aligned}
u_k &= 1 + \textstyle\sum_j \sum_{t=1}^{T} \varphi_{jtk}, \\
v_k &= \gamma + \textstyle\sum_j \sum_{t=1}^{T} \sum_{l=k+1}^{K} \varphi_{jtl}, \\
\lambda_{kw} &= \eta + \textstyle\sum_j \sum_{t=1}^{T} \varphi_{jtk} \left(\sum_n \zeta_{jnt} I[w_{jn} = w]\right),
\end{aligned}
$$

Natural gradient for λ

Details:

$$\log q(\phi|\lambda) = \lambda^T \log \phi - g(\lambda)$$

$$\mathcal{D}\mathcal{L}_j(\lambda) = (-\lambda + Ds_j)^T \mathbb{E}[\log \phi] + g(\lambda)$$

$$\frac{\partial g(\lambda)}{\partial \lambda} = \mathbb{E}[\log \phi]$$

$$g(\lambda) = \sum_w \log \Gamma(\lambda_w) - \log \Gamma(\sum_w \lambda_w)$$

$$\frac{\partial \mathcal{D}\mathcal{L}_j(\lambda)}{\partial \lambda} = \frac{\partial \mathbb{E}[\log \phi]}{\partial \lambda}(-\lambda + Ds_j)$$

$$\frac{\partial \mathbb{E}[\log \phi]}{\partial \lambda} = \frac{\partial^2 g(\lambda)}{\partial \lambda \partial \lambda^T} = -\frac{\partial^2 \log q(\phi|\lambda)}{\partial \lambda \partial \lambda^T} = -\mathbb{E}\left[\frac{\partial^2 \log q(\phi|\lambda)}{\partial \lambda \partial \lambda^T}\right]$$

Online Variational Updates

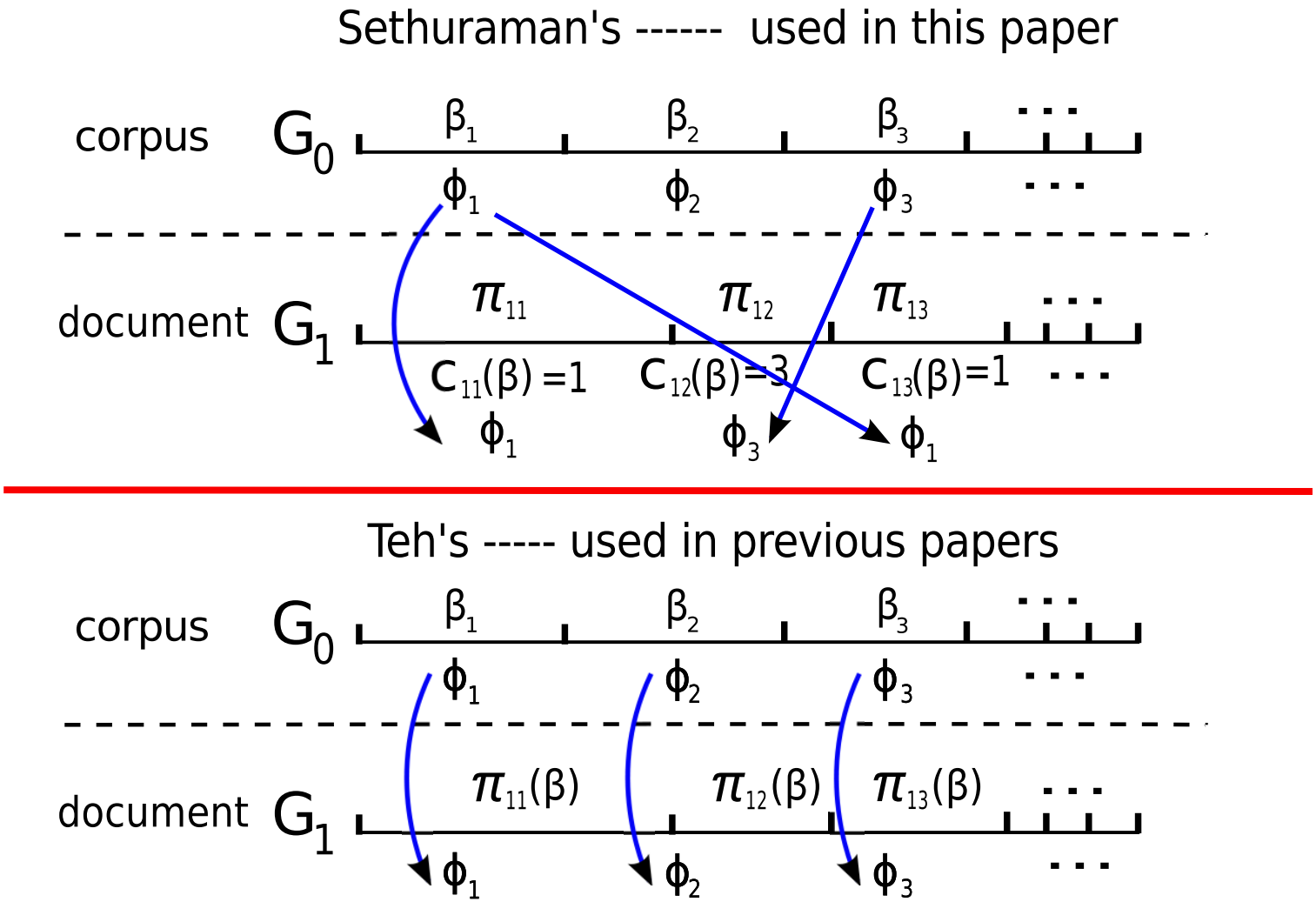
The online updates are

$$\begin{aligned}\boldsymbol{\lambda} &\leftarrow \boldsymbol{\lambda} + \rho_{t_o} \partial\boldsymbol{\lambda}(j),\\ \mathbf{u} &\leftarrow \mathbf{u} + \rho_{t_o} \partial\mathbf{u}(j)\\ \mathbf{v} &\leftarrow \mathbf{v} + \rho_{t_o} \partial\mathbf{v}(j),\end{aligned}$$

where

$$\begin{aligned}\partial\lambda_{kw}(j) &= -\lambda_{kw} + \eta + D\sum_{t=1}^{T}\varphi_{jtk}\left(\sum_n \zeta_{jnt} I[w_{jn} = w]\right),\\ \partial u_k(j) &= -u_k + 1 + D\sum_{t=1}^{T}\varphi_{jtk},\\ \partial v_k(j) &= -v_k + \gamma + D\sum_{t=1}^{T}\sum_{l=k+1}^{K}\varphi_{jtl}.\end{aligned}$$

Sethuraman's stick breaking for the HDP vs Teh's stick breaking



The online variational inference algorithm — cont'

For batch variational inference (review): optimize the lower bound $\mathcal{L}$ for entire data w.r.t. variational parameters. For example, for topic distribution $q(\phi|\lambda) = \text{Dirichlet}(\phi|\lambda)$, setting $\partial\mathcal{L}/\partial\lambda = 0$ gives,

$$\lambda \leftarrow \eta + \sum_{j=1}^{D} s_j(\lambda) \quad \text{— } s_j(\lambda) \text{ is the statistic from document } j.$$

For online variational inference, with appropriate learning rate ρ_t and document j sampled,

$$\lambda_{t+1} \leftarrow \lambda_t + \rho_t \boxed{A_t} \boxed{\frac{\partial D\mathcal{L}_j}{\partial \lambda_t}} \quad \text{— } A_t \text{ is a positive definite matrix.}$$

- If A_t is an identity matrix, it becomes stochastic gradient ascent.
- However, computing $\boxed{\frac{\partial D\mathcal{L}_j}{\partial \lambda_t}}$ is quite involved.

The online variational inference algorithm — cont'

Solution: given full conjugacy of the model and the closed-form updates, if we take (Amari [1998], Sato [2005]),

$$A_t = \left[-\frac{\partial^2 \log q(\phi|\lambda_t)}{\partial \lambda_t \partial \lambda_t^T} \right]^{-1}, \text{ the inverse of Riemannian metric of } q(\phi|\lambda)$$

$$\text{Then: } A_t \frac{\partial D\mathcal{L}_j}{\partial \lambda_t} = -\lambda_t + \eta + Ds_j(\lambda_t) \text{ — natural gradient!}$$

This results in a very similar structure as the coordinate updates as in the batch inference algorithm!

$$\lambda_{t+1} \leftarrow (1-\rho_t)\lambda_t + \rho_t \boxed{(\eta + Ds_j(\lambda_t))} \quad \text{— online variational inference.}$$

$$\lambda \leftarrow \boxed{\eta + \textstyle\sum_{j=1}^{D} s_j(\lambda)} \quad \text{— batch variational inference.}$$

Mini-batches: using use multiple samples each time.

Sethuraman's stick breaking for the HDP

Appy Sethuraman's stick breaking for document j, $G_j \sim \mathrm{DP}(\alpha_0 G_0)$ (Fox et al. [2008]),

$\boldsymbol{\pi}_j \sim \mathrm{GEM}(\alpha_0)$

$\psi_{jt} \sim G_0$, —— atoms/topics, the same set as in G_0, with duplications!

$G_j = \sum_{t=1}^{\infty} \pi_{jt} \delta_{\psi_{jt}}$.

The end. There is nothing left.