**Supplementary Material: Lexical and Hierarchical Topic Regression**

Viet-An Nguyen  
Computer Science  
University of Maryland  
College Park, MD  
vietan@cs.umd.edu

Jordan Boyd-Graber  
iSchool & UMIACS  
University of Maryland  
College Park, MD  
jbg@umiacs.umd.edu

Philip Resnik  
Linguistics & UMIACS  
University of Maryland  
College Park, MD  
resnik@umd.edu

| $D$ | # documents |
| $S_d$ | # sentences in document $d$ |
| $S_{d,t}$ | # groups (i.e., sentences) sitting at table $t$ in restaurant $d$ |
| $N_{d,s}$ | # tokens $w_{d,s}$ |
| $N_{d,l}$ | # tokens in $w_d$ assigned to level $l$ |
| $N_{d,>l}$ | # tokens in $w_d$ assigned to level $>l$ |
| $N_{d,\geq l}$ | $N_{d,l} + N_{d,>l}$ |
| $M_{c,l}$ | # tables at level $l$ on path $c$ |
| $C_{c,l,v}$ | # word type $v$ assigned to level $l$ on path $c$ |
| $C_{d,x,l,v}$ | # word type $v$ in $w_{d,x}$ assigned to level $l$ |
| $\phi_k$ | Topic at node $k$ |
| $\eta_k$ | Regression parameter at node $k$ |
| $\tau_v$ | Regression parameter of word type $v$ |
| $c_{d,t}$ | Table assignment for table $t$ in restaurant $d$ |
| $t_{d,s}$ | Table assignment for group $w_{d,s}$ |
| $z_{d,s,n}$ | Level assignment for $w_{d,s,n}$ |
| $k_{d,s,n}$ | Node assignment for $w_{d,s,n}$ (i.e., node at level $z_{d,s,n}$ on path $c_{d,t_{d,s}}$) |
| $L$ | Height of the tree |
| $C^+$ | Set of all possible paths (including new ones) of the tree |

Table 1: Notation used for SHLDA’s model and inference

This supplementary material provides more detail for the inference algorithm described in the main paper. First, we expand the two probabilities defined in Equations 1 and 2.

Equation 1 defines the conditional density of an arbitrary set of tokens $v_{d,x}$ (e.g., a token, a sentence or a set of sentences) in document $d$ being assigned to path $c$ given all other assignments.

$$ f^{-d,x}_c(v_{d,x}) = P(v_{d,x} | v^{-d,x}, c_{d,x}, c^{-d,x}, t, z) $$

$$ = \prod_{l=1}^{L} P(v_{d,x,l} | v^{-d,x}, c_{d,x}, c^{-d,x}, t, z) $$

$$ = \prod_{l=1}^{L} P(v_{d,x,l} | v^{-d,x,l}, c_{d,x}, c^{-d,x}, t, z) / P(v^{-d,x,l} | c_{d,x}, c^{-d,x}, t, z) $$

$$ = \prod_{l=1}^{L} \int P(v_{d,x,l}, v^{-d,x,l} | \phi_{c,l}) P(\phi_{c,l} | \beta_l) d\phi_{c,l} $$

$$ = \prod_{l=1}^{L} \int P(v_{d,x,l}, v^{-d,x,l} | \phi_{c,l}) \frac{\Gamma(C_{c,l}^{-d,x} + V \beta_l)}{\Gamma(C_{c,l}^{-d,x} + C_{d,x,l,v} + \beta_l)} \frac{\Gamma(C_{c,l}^{-d,x,v} + C_{d,x,l,v} + \beta_l)}{\Gamma(C_{c,l}^{-d,x,v} + \beta_l)} $$

(A.1)
where we use $v_{d,x,l}$ to denote the set of tokens in $v_{d,x}$ that are assigned to level $l$. $C_{c,l,v}$ is the number of times word type $v$ is assigned to node at level $l$ on path $c$. $C_{d,x,l,v}$ is the number of times word type $v$ in $v_{d,x}$ is assigned to node at level $l$ on path $c$. Superscript $-d,x$ denotes the same count excluding the assignments of $v_{d,x}$. Marginal counts are represented by $\cdot$.

Equation 2 defines the conditional density of the response variable $y_d$ of document $d$ given the set of tokens $v_{d,x}$ assigned to path $c$ and all other assignments

$$
g_{c}^{-d,x}(y_d) \equiv P(y_d | c_{d,x}, c^{-d,x}, z, t)$$

$$= \mathcal{N} \left( \frac{1}{N_{d,c}} \left( \sum_{d,x \in \{d\} \setminus v_{d,x}} \eta_{c,d,x,z_{d,s,n}} + \sum_{l=1}^{L} \eta_{c,l} \cdot C_{d,x,l} + \sum_{s=1}^{S_{d}} \sum_{n=1}^{N_{d,s}} \tau_{w_{d,s,n}} \right), \rho \right)$$

For new node at level $l$ on a new path $c^{new}$, we integrate over all possible values of $\eta_{c^{new},l}$ by using the following property of Gaussian distribution

$$\int \mathcal{N}(a + bx; y, \sigma_y) \mathcal{N}(y; \mu, \sigma_y) dy = \mathcal{N}(a + bx; \mu, b^2 \sigma_x + \sigma_y)$$

**Sampling t:** For each group (i.e., sentence) $w_{d,s}$, we need to sample a table $t_{d,s}$. The conditional distribution of a table $t$ in a Chinese restaurant process. For an existing table, this can be easily computed by multiplying Equation A.1 with Equation A.2. For a new table, we need to sum over all possible paths $c$ of the tree, including new ones.

$$P(t_{d,s} = t | rest) \propto P(t_{d,s} = t | t^{-d,-s}) \cdot P(w_{d,s}, y_d | t_{d,s} = t, w^{-d,-s}, t^{-d,-s}, z, c, \eta)$$

The second factor is the joint probability of observing $w_{d,s}$ and $y_d$ assigned to table $t_{d,s} = t$. If $t$ is an existing table, this can be easily computed by multiplying Equation A.1 with Equation A.2. For a new table, we need to sum over all possible paths $c$ of the tree, including new ones.

$$P(w_{d,s}, y_d | t_{d,s} = t, w^{-d,-s}, t^{-d,-s}, z, c, \eta)$$

$$\propto \left\{ \sum_{c \in C^+} P(c_{d,t^{new}} = c | c^{-d,-s}) \cdot \int_{s} f_{c,d}^{-d,s} w_{d,s} \cdot g_{c,d}^{-d,s} (y_d), \right\}$$

where $P(c_{d,t^{new}} = c | c^{-d,-s})$ is the prior probability of a path $c$, which is

$$P(c_{d,t^{new}} = c | c^{-d,-s}) \propto$$

$$\left\{ \prod_{l=2}^{L} \frac{M_{c,l}^{-d,s}}{M_{c,l-1}^{-d,s} + \gamma_{l-1}}, \right\}$$

$$\left\{ \gamma_{l} \cdot \prod_{l=2}^{L} \frac{M_{c,new,l}^{-d,s}}{M_{c,new,l-1}^{-d,s} + \gamma_{l-1}}, \right\}$$

Here we use $M_{c,l}$ to denote the number of tables assigned to node at level $l$ on path $c$. As usual, the superscript $-d,s$ denotes the same count but excluding assignments of $w_{d,s}$.

**Sampling z:** After assigning a sentence $w_{d,s}$ to a table, we assign each token $w_{d,s,n}$ to a level to choose a dish from the combo associated with the table. The probability of assigning $w_{d,s,n}$ to level $l$ conditioning on other assignments is

$$P(z_{d,s,n} = l | rest) \propto P(z_{d,s,n} = l | z^{-d,s,n}) \cdot P(w_{d,s,n}, y_d | z_{d,s,n} = l, w^{-d,s,n}, z^{-d,s,n}, t, c, \eta)$$

(A.5)
The first factor captures the probability that a customer in restaurant \( d \) is assigned to level \( l \), conditioned on the level assignments of all other customers in restaurant \( d \). Since the level distribution is modeled using a truncated stick breaking prior \( \text{GEM}(m, \pi) \), this probability is the posterior expected value of the \( l^\text{th} \) weight from the stick [1]

\[
P(z_{d,s,n} = l \mid z_{d,s,n}^-) = \frac{m\pi + N_{d,l}^{-d,s,n}}{\pi + N_{d,>l}^{-d,s,n}},
\]

where \( N_{d,l} \) is the number of tokens in document \( d \) assigned to level \( l \); \( N_{d,>l} \) is the number of tokens in document \( d \) assigned to level \( > l \); and \( N_{d,>l} = N_{d,l} + N_{d,>l} \).

The second factor is the probability of observing \( w_{d,s,n} \) and \( y_d \), conditioning on \( w_{d,s,n} \) being assigned to level \( l \) and other assignments. This is computed using Equations A.1 and A.2 as follow

\[
P(w_{d,s,n}, y_d \mid z_{d,s,n} = l, w_{d,s,n}^-, z_{d,s,n}^-, t, c, \eta) = f_{c_{d,t},d,s}(w_{d,s,n}) \cdot g_{c_{d,t},d,s}(y_d).
\]

**Sampling c**: After assigning customers to tables and levels, we also sample the path assignments for all tables. This is important since it potentially changes the assignments of all customers sitting at a given table, which leads to a well-mixed Markov chain and faster convergence. The probability of assigning a table \( t \) in restaurant \( d \) to a path \( c \) is

\[
P(c_{d,t} = c \mid \text{rest}) \propto P(c_{d,t} = c) \cdot P(w_{d,t}, y_d \mid c_{d,t} = c, w_{d,t}, c_{-d,t}, t, z, \eta) \tag{A.6}
\]

where we slightly abuse the notation by using \( w_{d,t} \equiv \bigcup_{s \mid t_{d,c} = t} w_{d,s} \) to denote the set of customers in all the groups sitting at table \( t \) in restaurant \( d \). The first factor is the prior probability of a path given all tables' path assignments \( c_{-d,t} \), excluding table \( t \) in restaurant \( d \) and is computed using Equation A.4.

The second factor in Equation A.6 is the probability of observing \( w_{d,t} \) and \( y_d \) given the new path assignments, \( P(w_{d,t}, y_d \mid c_{d,t} = c, w_{d,t}, c_{-d,t}, t, z, \eta) = f_{c_{d,t},d,t}(w_{d,t}) \cdot g_{c_{d,t},d,t}(y_d). \)

**Optimizing \( \eta \) and \( \tau \)**: We optimize the regression parameters \( \eta \) and \( \tau \) via the likelihood

\[
\mathcal{L}(\eta, \tau) = -\frac{1}{2\rho} \sum_{d=1}^{D} (y_d - \eta^T z_d - \tau^T \bar{w}_d)^2 - \frac{1}{2\sigma} \sum_{k=1}^{K} (\eta_k - \mu)^2 - \frac{1}{\omega} \sum_{v=1}^{V} |\tau_v|, \tag{A.7}
\]

The derivatives of this objective function with respect to each \( \eta_k \) is

\[
\frac{d\mathcal{L}(\eta, \tau)}{d\eta_k} = -\frac{1}{\rho} \sum_{d=1}^{D} \tilde{z}_{d,k} \cdot (\eta^T z_d + \tau^T \bar{w}_d - y_d) - \frac{1}{\sigma} (\eta_k - \mu)
\]

Since the L1-norm on \( \tau \) makes \( \mathcal{L}(\eta, \tau) \) non-differentiable when \( \tau_v = 0 \), we use the sub-gradient strategy [2] to approximate this gradient. Another heuristic to address this problem is to manually set the value of \( \tau_v \) equal to 0 for a subset of word types \( v \) in the vocabulary that have high TF-IDFs and perform L2-norm regularization on the remaining word types. We found that using this heuristic works reasonably well in practice and is able to speed up the optimization procedure significantly (depending on how large the set of word types that have \( \tau_v = 0 \)).

**References**
