A Variational Approximation for Topic Modeling of Hierarchical Corpora: Supplementary Material

Do-kyum Kim DOK027@CS.UCSD.EDU
Geoffrey M. Voelker VOELKER@CS.UCSD.EDU
Lawrence K. Saul SAUL@CS.UCSD.EDU
Department of Computer Science and Engineering, University of California, San Diego

S1. Proof of Theorem 3.1

The basic steps to prove theorem 3.1 are contained in two lemmas.

Lemma S1.1. *Let* $f(x) = \log\Gamma(x) + \log(x) - x\log(x)$. *Then* $f(x)$ *is a concave function of* $x > 0$.

Proof. We prove concavity by showing $f''(x) < 0$ for all $x > 0$. Taking derivatives, we find:

$$f''(x) = \Psi'(x) - \frac{1}{x^2} - \frac{1}{x}, \tag{S1}$$

where $\Psi(x)$ denotes the digamma function and $\Psi'(x)$ its derivative. A useful identity for this derivative (Abramowitz & Stegun, 1964) is the infinite series representation:

$$\Psi'(x) = \sum_{k=0}^{\infty} \frac{1}{(x+k)^2}. \tag{S2}$$

The lemma follows by substituting this series representation into eq. (S1). In particular, we have:

$$\begin{aligned} f''(x) &= -\frac{1}{x} + \sum_{k=1}^{\infty} \frac{1}{(x+k)^2} \\ &< -\frac{1}{x} + \frac{1}{x(x+1)} + \frac{1}{(x+1)(x+2)} + \cdots \\ &= -\frac{1}{x} + \left[\frac{1}{x} - \frac{1}{x+1}\right] + \left[\frac{1}{x+1} - \frac{1}{x+2}\right] + \cdots \\ &= 0 \end{aligned}$$

This completes the proof, but we gain more intuition by plotting $f(x)$ as shown in Fig. S1. Note that $\log\Gamma(x)$, which contains only the first term in $f(x)$, is a *convex* function of x. Thus it is the other terms in $f(x)$ that flip the sign of its second derivative. Essentially, the concavity of $f(x)$ is established by adding $\log x$ at small x and by subtracting $x\log x$ at large x. □

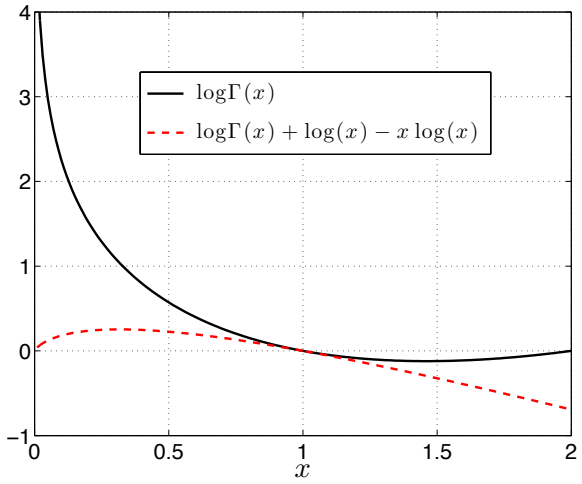

Figure S1. Plots of the *convex* function $\log\Gamma(x)$ and the *concave* function $\log\Gamma(x) + \log x - x\log x$ for $x > 0$.

Lemma S1.2. *Let* x *be a nonnegative random variable with bounded* $\mathrm{E}[\log(1/x)] < \infty$. *Then:*

$$\begin{aligned} \mathrm{E}[\log\Gamma(x)] \;\leq\; & \log\Gamma(\mathrm{E}[x]) + \log\mathrm{E}[x] - \mathrm{E}[\log x] \\ & + \mathrm{E}[x\log x] - \mathrm{E}[x]\log\mathrm{E}[x]. \end{aligned} \tag{S3}$$

Proof. Let $f(x)$ denote a concave function on $x > 0$. From Jensen's inequality, we have that $\mathrm{E}[f(x)] \leq f(\mathrm{E}[x])$. The result follows by setting $f(x) = \log\Gamma(x) + \log x - x\log x$ as in Lemma S1.1. □

Note that a naive application of Jensen's inequality to the left hand side of eq.(S3) yields the *lower* bound $\mathrm{E}[\log\Gamma(x)] \geq \log\Gamma(\mathrm{E}[x])$. Thus it is the additional terms on the right hand side of eq. (S3) that establish the *upper* bound. The direction of this inequality is crucial in the context of variational inference, where the upper bound in eq. (S3) is needed to maintain an

overall lower bound on the log-likelihood. Equipped with this lemma, we can now prove our main result.

Proof of Theorem 3.1. Let $\theta \sim \text{Dirichlet}(\nu)$, and also let $\alpha > 0$. Setting $x = \alpha\theta_i$ in eq. (S3) gives:

$$\begin{aligned}\mathrm{E}[\log\Gamma(\alpha\theta_i)] \;\leq\; &\log\Gamma(\alpha\mathrm{E}[\theta_i])) + \log\mathrm{E}[\theta_i] - \mathrm{E}[\log\theta_i] \\ &+ \alpha\mathrm{E}[\theta_i\log\theta_i] - \alpha\mathrm{E}[\theta_i]\log\mathrm{E}[\theta_i]. \quad \text{(S4)}\end{aligned}$$

All the expected values on the right hand side of this inequality can be computed analytically for Dirichlet random variables. In particular, let $\nu_0 = \sum_i \nu_i$. Then:

$$\mathrm{E}[\theta_i] = \frac{\nu_i}{\nu_0}, \quad \text{(S5)}$$

$$\mathrm{E}[\log\theta_i] = \Psi(\nu_i) - \Psi(\nu_0), \quad \text{(S6)}$$

$$\mathrm{E}[\theta_i\log\theta_i] = \mathrm{E}[\theta_i]\left(\mathrm{E}[\log\theta_i] + \frac{1}{\nu_i} - \frac{1}{\nu_0}\right). \quad \text{(S7)}$$

The theorem follows from substituting these statistics into eq. (S4). □

How tight is the bound in Lemma S1.2? The question is important because we use this inequality in conjunction with the variational approximation in eq. (3) to generate a lower bound on the log-likelihood. Here we make two useful observations.

First, we note that the bound in Lemma S1.2 is exquisitely tuned to the shape of the function $\log\Gamma(x)$ and the location of the expected value $\mathrm{E}[x]$. To see this, we provide an alternate derivation of the result in eq. (S3). We begin by appealing to the concavity of $f(x)$, established in Lemma S1.1; from this we obtain the upper bound

$$f(x) \;\leq\; f(x_0) + f'(x_0)(x - x_0), \quad \text{(S8)}$$

which holds for all values $x_0 > 0$. Now we recall the definition of $f(x)$ in Lemma S1.1 to obtain an upper bound on $\log\Gamma(x)$. Specifically we have:

$$\log\Gamma(x) = f(x) - \log x + x\log x, \quad \text{(S9)}$$

$$\leq f(x_0) + f'(x_0)(x - x_0) - \log x + x\log x. \quad \text{(S10)}$$

Figure S2 illustrates this upper bound on $\log\Gamma(x)$ for different values of x_0; note especially its tightness in the vicinity of x_0. The upper bound on $\mathrm{E}[\log\Gamma(x)]$ in eq. (S3) is based on choosing the best approximation from this family of upper bounds; it is easy to show that this occurs at $x_0 = \mathrm{E}[x]$. Thus we obtain the bound in Lemma S1.2 by taking expectations of both sides of eq. (S10) and setting $x_0 = \mathrm{E}[x]$.

Second, we note that the upper bound in eq. (S3) reduces to an equality in the limit of vanishing variance.

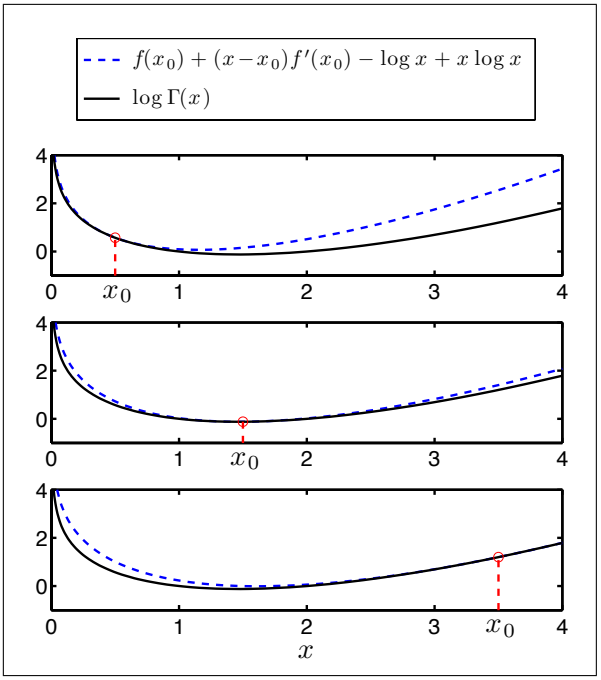

Figure S2. Tightness of the upper bound on $\log\Gamma(x)$ in eq. (S10) for different values of x_0.

In particular, this is the limit in which $\mathrm{E}[\log x] \to \log\mathrm{E}[x]$ and also $\mathrm{E}[x\log x] \to \mathrm{E}[x]\log\mathrm{E}[x]$. In this limit, the last four terms on the right hand side of eq. (S3) vanish, and we recover the result $\mathrm{E}[\log\Gamma(x)] = \log\Gamma(\mathrm{E}[x])$. In general, we expect factorized approximations such as eq. (3) to work well in the regime where the true posterior is peaked around its mean value. In this regime, we also expect the bound in eq. (S3) to be tight. Put another way, if it is sufficiently accurate to proceed with the factorized approximation in eq. (3), then we do not expect to incur much additional loss from the inequality in Lemma S1.2.

S2. Parallel Implementation of tiLDA

Here we briefly describe our scheme for parallelizing the recursive procedures in Algorithm 1. In practice, we obtain a significant speedup from this parallel implementation of tiLDA. This parallelization was necessary, for example, to obtain the results in section 4.

One naive manner of parallelization would simply be to allocate the inference for different top-level categories to different threads of execution. This approach, however, has two obvious limitations. First, inference in different categories may require different amounts of time; if the goal is to minimize idle CPU cycles, then we must more intelligently distribute the overall workload across different threads. Second, the number of

parallel threads at our disposal may greatly exceed the number of top-level categories. (For example, the BlackHatWorld corpus has only three top-level categories.) In this case, the naive approach to parallelization hardly makes the best use of available resources. In the following, we describe a parallel implementation of tiLDA that overcomes both these limitations.

Our parallel implementation is based on two key ideas. The first is to partition the algorithm into three types of tasks—START, DOCUMENT, and REPEAT—which we explain below. The second is to maintain a queue of these tasks and create multiple threads that execute tasks from this queue.

A START task is associated with every internal node in the corpus hierarchy. The task begins by initializing the node's parameters α_t and ν_t. After this initialization, the task then enqueues a new START task for each subcategory of the node and a DOCUMENT task for each document of the node. In Algorithm 1, the START task corresponds to lines 5–10.

A DOCUMENT task is associated with each document in the corpus. This task is responsible for optimizing the variational parameters ν_d and ρ_{dn} for documents given their observed words and (currently inferred) parameters of their parents. In Algorithm 1, the DOCUMENT task corresponds to the procedure called in line 10.

A REPEAT task is issued at each internal node in the corpus whenever all the tasks for the node's children complete. The REPEAT task is responsible for maximizing the lower bound on the log-likelihood $\mathscr{L}'$ with respect to the node's parameters. We mark the node as *complete* if the lower bound does not improve over its value from the previous REPEAT task at the node. Otherwise, we enqueue START and DOCUMENT tasks again for the node's children. The REPEAT task corresponds to executing lines 11–13 and then lines 6–10.

The overall algorithm begins with a START task at the root node and ends in a REPEAT task at the root node when the lower bound $\mathscr{L}'$ can no longer be improved.

S3. Background on Corpora

The Freelancer corpus collects seven years of job postings from Freelancer.com, one of the largest crowdsourcing sites on the Internet. The postings can be grouped by advertiser to form the three-level hierarchy shown in Fig. 1. In this hierarchy, tiLDA models the advertisers as second-level interior nodes and the job postings as third-level leaf nodes.

The BlackHatWorld corpus collects over two years of postings from the "BlackHatWorld" Internet forum. This data set was collected as part of a larger effort (Motoyama et al., 2011) to examine the social networks that develop in underground forums among distrustful parties. The BlackHatWorld forum evolved to discuss abusive forms of Internet marketing, such as bulk emailing (spam). The discussions are organized into the multi-level hierarchy shown in Fig. 2. We treat the threads in these subforums as documents for topic modeling. (We do not consider individual posts within threads as documents because they are quite short.)

We preprocessed these two corpora in the same way, removing stopwords from a standard list (Lewis et al., 2004), discarding infrequent words that appeared in fewer than 6 documents, and stemming the words that remain. In both data sets, we also pruned "barren" branches of the hierarchy: specifically, in the Freelancer corpus, we pruned advertisers with fewer than 20 job postings, and in the BlackHatWorld corpus, we pruned subforums with fewer than 60 threads.

S4. Additional Results

The multi-level tiLDA models can also be used to analyze hierarchical corpora in ways that go beyond the discovery of global topics. Recall that each tiLDA model yields topic proportions θ_t and a concentration parameter α_t for each category of the corpus. We can analyze these proportions and parameters for further insights into hierarchical corpora. In general, they provide a wealth of information beyond what can be discerned from (say) ordinary LDA.

Consider for example the Freelancer corpus. In this corpus, the categories of tiLDA represent advertisers, and the topic proportions of these categories can be used to profile the types of jobs that advertisers are trying to crowdsource. Summing these topic proportions over the corpus gives an estimate of the number of advertisers for each job type. Table S1 shows the results of this estimate: it appears that nearly one-third of advertisers on Freelancer.com are commissioning abuse-related jobs, and of these jobs, the majority appear to involve some form of SEO.

We gain further insights by analyzing the concentration parameters of individual advertisers. For example, the advertiser with the maximum concentration parameter ($\alpha_t = 4065.00$) on Freelancer.com commissioned 34 projects, among which 32 have nearly the exact same description. We also observe that advertisers with lower concentration parameters tend to be involved in a wider variety of projects.

Table S1. Estimated ratio of number of buyers in job types on the Freelancer data set.

Type	**Ratio**	**Type**	**Ratio**
SEO	18.47%	Affiliate Program	3.21%
Captcha Solving	2.68%	Account Creation	1.42%
Bulk Emailing	1.85%	OSN Linking	2.12%
Ad Posting	2.50%	Benign Jobs	67.74%

On the BlackHatWorld corpus, the topic proportions and concentration parameters of categories generally reflect the titles of their associated subforums. For example, the highest topic proportion (0.48) for "Email Marketing" belongs to the subforum on 'Email Marketing and Opt-In Lists,' and the highest topic proportion (0.59) for "Blogging" belongs to the 'Blogging' subforum. The highest concentration parameter (29.62) belongs to the 'Money, and Mo Money' subforum. This is not surprising as this subforum itself has only four subforums as children, all of which are narrowly focused on specific revenue streams; see Fig. 2.

References

Abramowitz, M. and Stegun, I. A. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Dover, New York, ninth dover printing, tenth GPO printing edition, 1964.

Lewis, D. D., Yang, Y., Rose, T. G., and Li, F. SMART stopword list. http://jmlr.csail.mit.edu/papers/volume5/lewis04a/a11-smart-stop-list/english.stop, April 2004.

Motoyama, M., McCoy, D., Levchenko, K., Savage, S., and Voelker, G. M. An Analysis of Underground Forums. In *Proceedings of the 2011 ACM SIGCOMM Internet Measurement Conference (IMC)*, pp. 71–80, 2011.