Combinatorial Topic Models using Small-Variance Asymptotics

Ke Jiang
Ohio State University

Suvrit Sra
MIT

Brian Kulis
Boston University

Abstract

Modern topic models typically have a probabilistic formulation, and derive their inference algorithms based on Latent Dirichlet Allocation (LDA) and its variants. In contrast, we approach topic modeling via combinatorial optimization, and take a small-variance limit of LDA to derive a new objective function. We minimize this objective by using ideas from combinatorial optimization, obtaining a new, fast, and high-quality topic modeling algorithm. In particular, we show that our results are not only significantly better than traditional SVA algorithms, but also truly competitive with popular LDA-based approaches; we also discuss the (dis)similarities between our approach and its probabilistic counterparts.

1 Introduction

Topic modeling has become a cornerstone of unsupervised learning on large document collections. While the roots of topic modeling date back to latent semantic indexing (Deerwester *et al.*, 1990) and probabilistic latent semantic indexing (Hofmann, 1999), the arrival of Latent Dirichlet Allocation (LDA) (Blei *et al.*, 2003) was a turning point that transformed the community's thinking about topic modeling. LDA led to several followups that address some limitations of the original model (Blei and Lafferty, 2006; Wang and Grimson, 2007), while paving the way for subsequent advances in Bayesian learning, including variational inference methods (Teh *et al.*, 2006b), nonparametric Bayesian models (Blei *et al.*, 2004; Teh *et al.*, 2006a), among others.

The LDA family of topic models are almost exclusively cast as probabilistic models. Consequently, the vast majority of techniques developed for topic modeling—collapsed Gibbs sampling (Griffiths and Steyvers, 2004), variational methods (Blei *et al.*, 2003; Teh *et al.*, 2006b), and "factorization" approaches with theoretical guarantees (Anandkumar *et al.*, 2012)—are centered around performing inference for underlying probabilistic models. By limiting ourselves to a purely probabilistic viewpoint, we may be missing opportunities arising from combinatorial thinking. This observation underlies our central question: *Can one obtain a combinatorial topic model that competes with LDA?*

We answer this question positively by proposing a new combinatorial formulation obtained via *small-variance asymptotics* (SVA) on the LDA model. SVA produces limiting versions of several probabilistic learning models, which can then be solved as combinatorial optimization problems. (A helpful analogy here is the relation between k-means and Gaussian mixtures as variance goes to zero.) Indeed, SVA techniques have been quite fruitful recently (Campbell *et al.*, 2013; Broderick *et al.*, 2013; Wang and Zhu, 2014), and a common theme is that computational gains and good empirical performance of k-means carry over to richer SVA based models.

But merely using SVA to obtain a combinatorial topic model is insufficient: we also need effective algorithms to optimize the model. A direct application of the popular greedy combinatorial procedures on the LDA-based SVA model *fails* to compete with the main probabilistic LDA methods. This setback necessitates a new idea. Surprisingly, as we will see, an improved word assignment technique combined with an incremental refinement procedure transforms the SVA approach into a competitive topic modeling algorithm.

Contributions. The main contributions of our paper are:

- We perform SVA on the standard LDA model to obtain a new combinatorial topic model.
- We develop algorithms for optimizing this combinatorial model by using ideas from facility location and incremental refinement. Moreover, we show how our procedure can be implemented to take just $O(NK)$ time per iteration to assign words to topics, where N is the total number of words and K the number of topics.

We demonstrate that our approach not only improves significantly over the traditional SVA algorithms, but also competes favorably with existing state-of-the-art topic modeling algorithms; in particular, our approach is orders of magnitude faster than sampling-based approaches, with comparable accuracy. Moreover, we show that the sampler's

Appearing in Proceedings of the 20^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2017, Fort Lauderdale, Florida, USA. JMLR: W&CP volume 54.

mixing time improves substantially when initialized using our combinatorial method for just a few iterations. We also compare against several recent theoretically-motivated algorithms (Anandkumar *et al.*, 2012; Podosinnikova *et al.*, 2015; Arora *et al.*, 2013) and variational inference methods.

Note. Before proceeding further, we note here an important point regarding evaluation of topic models. The connection between our approach and standard LDA may be viewed analogously to the connection between k-means and a Gaussian mixture model. As such, evaluation is nontrivial; most topic models are evaluated using predictive log-likelihood or related measures. In light of the "hard-vs-soft" analogy, a predictive log-likelihood score can be misleading for evaluating performance of the k-means algorithm, so clustering comparisons typically focus on ground-truth accuracy (when possible). Due to the lack of available ground truth data, to assess our combinatorial model we must resort to synthetic data sampled from the LDA model to enable meaningful quantitative comparisons; but in line with common practice we also present results on real-world data, for which we use both discrete and held-out log-likelihoods.

1.1 Related Work

LDA Algorithms. Many techniques have been developed for efficient inference for LDA. MCMC-based methods are quite popular, notably the collapsed Gibbs sampler (CGS) (Griffiths and Steyvers, 2004), and variational inference methods (Blei *et al.*, 2003; Teh *et al.*, 2006b). Among MCMC and variational techniques, CGS typically yields excellent results and is guaranteed to sample from the desired posterior with sufficiently many samples. However, it can be slow and many samples may be required before mixing.

Since topic models are often used on large (document) collections, significant effort has been made in scaling up LDA algorithms. One recent example is (Li *et al.*, 2014) that presents a massively distributed implementation. Such methods are outside the focus of this paper, where the emphasis is on a new combinatorial model that can quatitatively compete with the probabilistic LDA approaches. Ultimately, our model should be amenable to fast distributed solvers, and obtaining such solvers is an important part of future work.

A complementary line of algorithms starts with (Arora *et al.*, 2012, 2013), who consider certain separability assumptions on the input data to circumvent NP-Hardness of the basic LDA model. These works have shown performance competitive to Gibbs sampling in some scenarios while also featuring theoretical guarantees. Other recent viewpoints on LDA are offered by (Anandkumar *et al.*, 2012; Bansal *et al.*, 2014; Podosinnikova *et al.*, 2015).

Small-Variance Asymptotics (SVA). As noted above, SVA has recently emerged as an effective tool for obtaining scalable algorithms and objective functions by "hardening" probabilistic models. Similar connections are known for instance in dimensionality reduction (Roweis, 1997), multi-view learning, classification (Tong and Koller, 2000), and structured prediction (Samdani *et al.*, 2014). Starting with Dirichlet process mixtures (Kulis and Jordan, 2012), one thread of research has considered applying SVA to richer Bayesian nonparametric models. Applications include clustering (Kulis and Jordan, 2012), feature learning (Broderick *et al.*, 2013), evolutionary clustering (Campbell *et al.*, 2013), infinite hidden Markov models (Roychowdhury *et al.*, 2013), Markov jump processes (Huggins *et al.*, 2015), infinite SVMs (Wang and Zhu, 2014), and hierarchical clustering methods (Lee and Choi, 2015). A related thread of research considers applying SVA when the data likelihood is not Gaussian, which is precisely the scenario under which LDA falls. (Jiang *et al.*, 2012) show how SVA may be applied when the likelihood is a member of the exponential family and they consider topic modeling as a potential application, but no quantitative comparisons were provided.

However, almost all the algorithms proposed in the prior SVA literature share the same greedy local assignment step, which is known to succeed only under certain circumstances and often fails to minimize the objective greatly under random initialization (Yen *et al.*, 2015). We will show in the experiments that the popular greedy SVA algorithm (which we denote in this work as the `Basic` algorithm) fails to work well on topic models; the present paper fixes this issue by using a stronger word assignment algorithm and introducing incremental refinement.

Combinatorial Optimization. In developing effective algorithms for topic modeling, we will borrow some ideas from the large literature on combinatorial optimization algorithms. In particular, in the k-means community, significant effort has been made on how to improve upon the basic k-means algorithm, which is known to be prone to local optima; these techniques include local search methods (Dhillon *et al.*, 2002) and good initialization strategies (Arthur and Vassilvitskii, 2007). We also borrow ideas from approximation algorithms, most notably algorithms based on the facility location problem (Jain *et al.*, 2003).

2 SVA for Latent Dirichlet Allocation

We now detail our combinatorial approach to topic modeling. We start with the derivation of the underlying objective function that is the basis of our work. This objective is derived from the LDA model by applying SVA, and contains two terms. The first is similar to the k-means clustering objective in which it seeks to assign words to topics that are, in a particular sense, "close." The second term, arising from the Dirichlet prior on the per-document topic distributions, places a constraint on the number of topics per document in which it tries to exploit the word co-occurrence information.

Recall the standard LDA model. We choose topic weights for each document as $\theta_j \sim \text{Dir}(\alpha)$, where $j \in \{1, ..., M\}$.

Then we choose word weights for each topic as $\psi_t \sim \text{Dir}(\beta)$, where $t \in \{1, ..., K\}$. Then, for each word i in document j, we choose a topic $z_{ji} \sim \text{Cat}(\theta_j)$ and a word $w_{ji} \sim \text{Cat}(\psi_{z_{ji}})$. Here α and β are scalars (i.e., we are using a symmetric Dirichlet distribution). Let $\mathbf{W}$ denote the vector of all words in all documents, $\mathbf{Z}$ the topic indicators of all words in all documents, $\boldsymbol{\theta}$ the concatenation of all the θ_j variables, and $\boldsymbol{\psi}$ the concatenation of all the ψ_t variables. Also let N_j be the total number of word tokens in document j. The θ_j vectors are each of length K, the number of topics. The ψ_t vectors are each of length V, the size of the vocabulary. We can write down the full joint likelihood $p(\mathbf{W}, \mathbf{Z}, \boldsymbol{\theta}, \boldsymbol{\psi}|\alpha, \beta)$ of the model in the factored form

$$\prod_{t=1}^{K} p(\psi_t|\beta) \prod_{j=1}^{M} p(\theta_j|\alpha) \prod_{i=1}^{N_j} p(z_{ji}|\theta_j) p(w_{ji}|\psi_{z_{ji}}),$$

where each of the probabilities is as specified by the LDA model. We can eliminate variables to simplify inference by integrating out $\boldsymbol{\theta}$ to obtain

$$p(\mathbf{Z}, \mathbf{W}, \boldsymbol{\psi}|\alpha, \beta) = \int_{\boldsymbol{\theta}} p(\mathbf{W}, \mathbf{Z}, \boldsymbol{\theta}, \boldsymbol{\psi}|\alpha, \beta) d\boldsymbol{\theta}. \quad (1)$$

After integration and some simplification, (1) becomes

$$\left[\prod_{t=1}^{K} p(\psi_t|\beta) \prod_{j=1}^{M} \prod_{i=1}^{N_j} p(w_{ji}|\psi_{z_{ji}})\right] \times \left[\prod_{j=1}^{M} \frac{\Gamma(\alpha K)}{\Gamma(\sum_{t=1}^{K} n_{j\cdot}^{t} + \alpha K)} \prod_{t=1}^{K} \frac{\Gamma(n_{j\cdot}^{t} + \alpha)}{\Gamma(\alpha)}\right]. \quad (2)$$

Here $n_{j\cdot}^{t}$ is the number of word tokens in document j assigned to topic t. Following (Broderick *et al.*, 2013), we can obtain the SVA objective by taking the (negative) logarithm of this likelihood and letting the variance go to zero. Given space considerations, we will summarize this derivation; full details are available in the supplementary material.

Consider the first bracketed term of (2). Taking logs yields a sum over terms of the form $\log p(\psi_t|\beta)$ and terms of the form $\log p(w_{ji}|\psi_{z_{ji}})$. Noting that the latter of these is a categorical distribution, and thus a member of the exponential family, we can appeal to the results in (Banerjee *et al.*, 2005; Jiang *et al.*, 2012) to introduce a new parameter for scaling the variance. In particular, we can write $p(w_{ji}|\psi_{z_{ji}})$ in its *Bregman divergence* form $\exp(-\text{KL}(\tilde{w}_{ji}, \psi_{z_{ji}}))$, where KL refers to the discrete KL-divergence, and $\tilde{w}_{ji}$ is an indicator vector for the word at token w_{ji}. It is straightforward to verify that $\text{KL}(\tilde{w}_{ji}, \psi_{z_{ji}}) = -\log \psi_{z_{ji}, w_{ji}}$. Next, introduce a new parameter η that scales the variance appropriately, and write the resulting distribution as proportional to $\exp(-\eta \cdot \text{KL}(\tilde{w}_{ji}, \psi_{z_{ji}}))$. As $\eta \to \infty$, the expected value of the distribution remains fixed while the variance goes to zero, exactly what we require.

After this, consider the second bracketed term of (2). We scale α appropriately as well; this ensures that the hierarchical form of the model is retained asymptotically. In particular, we write $\alpha = \exp(-\lambda \cdot \eta)$. After some manipulation of this distribution, we can conclude that the negative log of the Dirichlet multinomial term becomes asymptotically $\eta\lambda(K_{j+} - 1)$, where K_{j+} is the number of topics currently used by document j. (The maximum value for K_{j+} is K, the total number of topics.) To formalize, let $f(x) \sim g(x)$ denote that $f(x)/g(x) \to 1$ as $x \to \infty$. Then we have the following (see the supplement for a proof):

Lemma 1. *Consider the likelihood*

$$p(\boldsymbol{Z}|\alpha) = \left[\prod_{j=1}^{M} \frac{\Gamma(\alpha K)}{\Gamma(\sum_{t=1}^{K} n_{j\cdot}^{t} + \alpha K)} \prod_{t=1}^{K} \frac{\Gamma(n_{j\cdot}^{t} + \alpha)}{\Gamma(\alpha)}\right].$$

If $\alpha = \exp(-\lambda \cdot \eta)$, then asymptotically as $\eta \to \infty$, the negative log-likelihood satisfies

$$-\log p(\boldsymbol{Z}|\alpha) \sim \eta\lambda \sum_{j=1}^{M} (K_{j+} - 1).$$

Now we put the terms of the negative log-likelihood together. The $-\log p(\psi_t|\beta)$ terms vanish asymptotically since we are not scaling β (see the note below on scaling β). Thus, the remaining terms in the SVA objective are the ones arising from the word likelihoods and the Dirichlet-multinomial. Using the Bregman divergence representation with the additional η parameter, we conclude that the negative log-likelihood asymptotically yields the following:

$$-\log p(\mathbf{Z}, \mathbf{W}, \boldsymbol{\psi}|\alpha, \beta) \sim \eta\left[\sum_{j=1}^{M} \sum_{i=1}^{N_j} \text{KL}(\tilde{w}_{ji}, \psi_{z_{ji}}) + \lambda \sum_{j=1}^{M} (K_{j+} - 1)\right],$$

which leads to our final objective function

$$\min_{\boldsymbol{Z}, \boldsymbol{\psi}} \left(\sum_{j=1}^{M} \sum_{i=1}^{N_j} \text{KL}(\tilde{w}_{ji}, \psi_{z_{ji}}) + \lambda \sum_{j=1}^{M} K_{j+}\right). \quad (3)$$

We remind the reader that $\text{KL}(\tilde{w}_{ji}, \psi_{z_{ji}}) = -\log \psi_{z_{ji}, w_{ji}}$. Thus, we obtain a k-means-like term that says that any word should be "close" to its assigned topic in terms of KL-divergence under the word co-occurrence constraint enforced with reasonable λ value. Notice that (3) reduces to the document-level K-means problem with $\lambda \to \infty$, and the token-level K-means with $\lambda \to 0$.

Note. We did not scale β to obtain a simpler objective with only one parameter (other than the total number of topics), but let us say a few words about scaling β. A natural approach is to further integrate out $\boldsymbol{\psi}$ of the joint likelihood, as is done in the collapsed Gibbs sampler. One would obtain additional Dirichlet-multinomial distributions, and properly scaling as discussed above would yield a simpler objective

Algorithm 1 Basic Batch Algorithm

Input: Words: $\mathbf{W}$, num. of topics: K, Topic penalty: λ
Initialize $\mathbf{Z}$ and topic vectors $\psi_1, ..., \psi_K$.
Compute initial objective function (3) using $\mathbf{Z}$ and $\boldsymbol{\psi}$.
repeat
  //Update assignments:
  for every word token i in every document j **do**
    Compute distance $d(j, i, t)$ to topic i: $-\log(\psi_{t,w_{ji}})$.
    If $z_{ji} \neq t$ for all tokens i in document j, add λ to $d(j, i, t)$.
    Obtain assignments via $Z_{ji} = \text{argmin}_t d(j, i, t)$.
  end for
  //Update topic vectors:
  for every element ψ_{tw} **do**
    ψ_{tw} = # occ. of word w in topic t / total # of word tokens in topic t.
  end for
  Recompute objective (3) using updated $\mathbf{Z}$ and $\boldsymbol{\psi}$.
until no change in objective function.
Output: Assignments $\mathbf{Z}$.

that places penalties on the number of topics per document as well as the number of words in each topic. Optimization would then be performed with respect to the topic assignment matrix. Future work will consider effectiveness of such an objective function for topic modeling.

3 Algorithms

With our combinatorial objective in hand, we are ready to develop algorithms that optimize it. First, we discuss a locally-convergent algorithm similar to k-means and the hard topic modeling algorithm (Jiang *et al.*, 2012). Then, we introduce two more powerful techniques: (i) a word-level assignment method that arises from connections between our proposed objective function and the facility location problem; and (ii) an incremental topic refinement method. Despite the apparent complexity of our algorithms, we show that the per-iteration time matches that of the collapsed Gibbs sampler (while empirically converging in just a few iterations, as opposed to the thousands typically required for Gibbs sampling).

Algorithm 1 shows the basic iterative algorithm, which follows a strategy similar to DP-means (Kulis and Jordan, 2012) and the hard topic modeling algorithm (Jiang *et al.*, 2012)—we perform alternate optimization by first minimizing with respect to the topic indicators for each word (the $\mathbf{Z}$ values) and then minimizing with respect to the topics (the $\boldsymbol{\psi}$ vectors). As with DP-means, the updates for $\mathbf{Z}$ cannot be computed exactly, due to the extra penalty term in the objective, so the basic algorithm uses a simple heuristic for the updates on $\mathbf{Z}$. The resulting algorithm has the advantage that it achieves local convergence. However, it works only under careful initializations, analogous to DP-means.

Algorithm 2 Improved Word Assignments for $\mathbf{Z}$

Input: Words: $\mathbf{W}$, num. of topics: K, Topic penalty: λ, Topics: $\boldsymbol{\psi}$
for every document j **do**
  Let $f_t = \lambda$ for all topics t.
  Initialize all word tokens to be unmarked.
  while there are unmarked tokens **do**
    Pick the topic t and set of unmarked tokens $\mathcal{W}$ that minimizes (4).
    Let $f_t = 0$ and mark all tokens in $\mathcal{W}$.
    Assign $z_{ji} = t$ for all $i \in \mathcal{W}$.
  end while
end for
Output: Assignments $\mathbf{Z}$.

3.1 Improved Word Assignments

In this section, we discuss and analyze an alternative assignment technique for $\mathbf{Z}$, which may be used as an initialization to the locally-convergent basic algorithm or to replace it completely.

Algorithm 2 details the alternate assignment strategy for tokens. The inspiration for this greedy algorithm arises from the fact that we can view the assignment problem for $\mathbf{Z}$, given $\boldsymbol{\psi}$, as an instance of the uncapacitated facility location (UFL) problem (Jain *et al.*, 2003). Recall that the UFL problem aims to open a set of facilities from a set F of potential locations. Given a set of clients D, a distance function $d : D \times F \to \mathbb{R}_+$, and a cost function $f : F \to \mathbb{R}_+$ for the set F, the UFL problem aims to find a subset S of F that minimizes $\sum_{i \in S} f_i + \sum_{j \in D} (\min_{i \in S} d_{ij})$.

To map UFL to the assignment problem in combinatorial topic modeling, consider the problem of assigning word tokens to topics for some fixed document j. The topics correspond to the facilities and the clients correspond to word tokens. Let $f_t = \lambda$ for each facility, and let the distances between clients and facilities be given by the corresponding KL-divergences as detailed earlier. Then the UFL objective corresponds exactly to the assignment problem for topic modeling. Algorithm 2 is a greedy algorithm for UFL that has been shown to achieve constant factor approximation guarantees when distances between clients and facilities forms a metric (Jain *et al.*, 2003) (this guarantee does not apply in our case, as KL-divergence is not a metric).

The algorithm, must select, among all topics and all unmarked tokens $\mathcal{W}$, the minimizer to

$$\frac{f_t + \sum_{i \in \mathcal{W}} \text{KL}(\tilde{w}_{ji}, \psi_t)}{|\mathcal{W}|}. \tag{4}$$

This algorithm appears to be computationally expensive, requiring multiple rounds of marking where each round requires us to find a minimizer over exponentially-sized sets.

Algorithm 3 Incremental Topic Refinements for $\mathbf{Z}$

Input: Words: $\mathbf{W}$, num. of topics: K, Topic penalty: λ, Assignment: $\mathbf{Z}$, Topics: $\boldsymbol{\psi}$
randomly permute the documents.
for every document j **do**
  for each mini-topic S, where $z_{js} = t \; \forall s \in S$ for some topic t **do**
    for every other topic $t' \neq t$ **do**
      Compute $\Delta(S, t, t')$, the change in the obj. function when re-assigning $z_{js} = t' \; \forall s \in S$.
    end for
    Let $t^* = \text{argmin}_{t'} \Delta(S, t, t')$.
    Reassign tokens in S to t^* if it yields a smaller obj.
    Update topics $\boldsymbol{\psi}$ and assignments $\mathbf{Z}$.
  end for
end for
Output: Assignments $\mathbf{Z}$ and Topics $\boldsymbol{\psi}$.

Surprisingly, under mild assumptions we can use the structure of our problem to derive an efficient implementation of this algorithm that runs in total time $O(NK)$. The details of this efficient implementation are presented in the supplementary material.

3.2 Incremental Topic Refinement

In this section, we try to refine the results exploring the hierarchical structure in topic modeling: we have both word-level assignments and "mini-topics" (formed by word tokens in the same document which are assigned to the same topic). Explicitly refining the mini-topics should help in achieving better word-coassignment within the same document. This can be considered as analogously to the block coordinate descent algorithm (Bertsekas, 1999) in the continuous optimization and is also similar to the local search techniques in the clustering literature (Dhillon *et al.*, 2002).

More specifically, we consider an incremental topic refinement scheme that works as follows. For a given document, we consider swapping all word tokens assigned to the same topic within that document to another topic. We compute the change in objective function that would occur if we updated the topic assignments for those tokens and then updated the resulting topic vectors. Specifically, for document j and its mini-topic S formed by its word tokens assigned to topic t, the objective function change can be computed by

$$\begin{aligned}\Delta(S, t, t') = &- (n^t_{\cdot\cdot} - n^t_{j\cdot})\phi(\boldsymbol{\psi}^-_t) - (n^{t'}_{\cdot\cdot} + n^t_{j\cdot})\phi(\boldsymbol{\psi}^+_{t'}) \\ &+ n^t_{\cdot\cdot}\phi(\boldsymbol{\psi}_t) + n^{t'}_{\cdot\cdot}\phi(\boldsymbol{\psi}_{t'}) - \lambda \mathbb{I}[t' \in \mathcal{T}_j],\end{aligned}$$

where $n^t_{j\cdot}$ is the number of tokens in document j assigned to topic t, $n^t_{\cdot\cdot}$ is the total number of tokens assigned to topic t, $\boldsymbol{\psi}^-_t$ and $\boldsymbol{\psi}^+_{t'}$ are the updated topics, $\mathcal{T}_j$ is the set of all the topics used in document j, and $\phi(\boldsymbol{\psi}_t) = \sum_w \psi_{tw} \log \psi_{tw}$.

We accept the move if $\min_{t' \neq t} \Delta(S, t, t') < 0$ and update the topics $\boldsymbol{\psi}$ and assignments $\mathbf{Z}$ accordingly. Then we continue to the next mini-topic, hence the term "incremental". Since $\boldsymbol{\psi}$ and $\mathbf{Z}$ are updated in every objective-decreasing move, we randomly permute the processing order of the documents in each iteration. This usually helps in obtaining better results in practice. See Algorithm 3 for details.

At first glance, it appears that this incremental topic refinement strategy may be computationally expensive. However, computing the global change in objective function $\Delta(S, t, t')$ can be computed in $O(|S|)$ time, if the topics are maintained by count matrices. Only the counts involving the words in the mini-topic and the total counts are affected. Since we compute the change across all topics, and across all mini-topics S, the total running time of the incremental topic refinement is $O(NK)$, as for the basic batch algorithm and the improved word assignment algorithm.

4 Experiments

In this section, we compare the algorithms proposed above with the basic algorithm and their probabilistic counterparts.

4.1 Synthetic Documents

Our first set of experiments is on simulated data. We compare two versions of our algorithms—Improved Word Assignment (`Word`), and Improved Word with Topic Refinement (`Word+Refine`)—with the traditional Basic Batch algorithm (`Basic`), the collapsed Gibbs sampler (`CGS`) (Griffiths and Steyvers, 2004), the standard variational inference algorithm (`VB`) (Blei *et al.*, 2003), the spectral algorithm (`Spectral`) (Anandkumar *et al.*, 2012), the orthogonal joint diagonalization (`JD`) (Podosinnikova *et al.*, 2015), the tensor power method (`TPM`) (Podosinnikova *et al.*, 2015) and the `Anchor` method[1] (Arora *et al.*, 2013).

Methodology. Due to a lack of ground truth data for topic modeling, we benchmark on synthetic data. We train all algorithms on the following data sets. (A) documents sampled from an LDA model with $\alpha = 0.04, \beta = 0.05$, with 20 topics and having vocabulary size 2000. Each document has length 150. (B) documents sampled from an LDA model with $\alpha = 0.02, \beta = 0.01$, 50 topics and vocabulary size 3000. Each document has length 200.

For the collapsed Gibbs sampler, we collect 10 samples with 30 iterations of thinning after 3000 burn-in iterations. The variational inference runs for 100 iterations. The `Word` algorithm replaces basic word assignment with the improved word assignment step within the batch algorithm, and `Word+Refine` further alternates between improved word and incremental topic refinement steps. The `Word` and `Word+Refine` are run for 20 and 10 iterations, respectively. All the algorithms are initialized by randomly assigning each word to one of the topics, whenever applicable. For `Basic`, `Word` and `Word+Refine`, we run experiments with $\lambda \in \{6, 7, 8, 9, 10, 11, 12\}$, and the best

[1] All the codes used are provided by the authors.

SynthA	$\lambda = 8$	$\lambda = 9$	$\lambda = 10$	$\lambda = 11$	$\lambda = 12$
Basic	0.027 / 0.009	0.027 / 0.009	0.027 / 0.009	0.027 / 0.009	0.027 / 0.009
Word	0.724 / 0.669	0.730 / 0.660	0.786 / 0.750	0.786 / 0.745	0.784 / 0.737
Word+Refine	0.828 / 0.838	0.839 / 0.850	0.825 / 0.810	0.847/ **0.859**	**0.848 / 0.859**
CGS	0.829 / 0.839				
SynthB	$\lambda = 6$	$\lambda = 7$	$\lambda = 8$	$\lambda = 9$	$\lambda = 10$
Basic	0.043 / 0.007	0.043 / 0.007	0.043 / 0.007	0.043 / 0.007	0.043 / 0.007
Word	0.850 / 0.737	0.854 / 0.743	0.855 / 0.752	0.855 / 0.750	0.850 / 0.743
Word+Refine	0.922 / 0.886	**0.926 / 0.901**	0.913 / 0.860	0.923 / 0.899	0.914 / 0.876
CGS	0.917 / 0.873				

Table 1: The NMI scores and Adjusted Rand Index (NMI/ARand, best results in bold, higher is better) for word assignments of our algorithms for both synthetic datasets with 5000 documents.

results are presented if not stated otherwise. In contrast, the *true* α, β parameters are provided as inputs to the CGS, VB, Spectral, JD and TPM algorithms. We note that we are heavily handicapped by this setup, since these algorithms are designed specifically for data from the LDA model. Here, we use LDA moments (Anandkumar *et al.*, 2012) for the Spectral, JD and TPM algorithms. Please see the supplementary material for results with the discrete independent analysis cumulants (Podosinnikova *et al.*, 2015) and varied document length datasets.

Assignment accuracy. Both the Gibbs sampler and our algorithms provide word-level topic assignments. Thus we can compare the training accuracy of these assignments, which is shown in Table 1. The result of the Gibbs sampler is given by the *highest* among all the samples selected. The accuracy is shown in terms of the *normalized mutual information* (NMI) score and the *adjusted Rand index* (ARand), which are both in the range of [0,1] and are standard evaluation metrics for clustering problems.

Despite the similarity with the Gibbs sampler, we can see that the Basic algorithm, which has the same assignment update strategy as with existing SVA algorithms, performs poorly. This shows that the Basic algorithm is very sensitive to the initialization and the λ value. Unlike Basic, the Word algorithm greatly boosts the assignment accuracy. With further help from topic refinement, we match or marginally exceed the performance of the Gibbs sampler for a wide range of λ.

Topic reconstruction error. Now we look at the reconstruction error between the true topic-word distributions and the learned distributions. In particular, given a learned topic matrix $\hat{\psi}$ and the true matrix ψ, we use the Hungarian algorithm (Kuhn, 1955) to align topics, and then evaluate the ℓ_1 distance between each pair of topics. Table 2 presents the mean reconstruction errors per topic of different learning algorithms for varying number of documents. As a baseline, we also include the results from the k-means algorithm with KL-divergence (Banerjee *et al.*, 2005) where each document is assigned to a single topic.

Among the three proposed algorithms, similar to the situation above, the Basic algorithm performs the worst in all the data settings, even worse than the k-means algorithm. The topic refinement step provides a significant improvement, which helps to reduce the ℓ_1 error at least 60% from the Word algorithm only. The Gibbs sampler has the lowest ℓ_1 on smaller corpora, where Word+Refine and Anchor come next. However, for the larger corpora, the sampler needs to run much longer to reach a lower ℓ_1 error, and can not compete with Word+Refine and Anchor for even 3000 iterations. We again want to emphasize here that CGS, VB, Spectral, JD and TPM[2] are given the *true* parameters as input.

As observed above, the Gibbs sampler can easily become trapped in a local optima area and needs many iterations on large data sets, which can be seen from Figure 1. Since our algorithm outputs $\mathbf{Z}$, we can use this assignment as initialization to the sampler. In Figure 1, we show the evolution of topic reconstruction ℓ_1 error initialized with the Word+Refine optimized assignment for only 3 iterations with varying values of λ. With these semi-optimized initializations, we observe more than a 5-fold speed-up compared to random initializations with no special choice of λ.

Running Time. For our current implementation, an iteration of Refine is roughly equivalent to one Gibbs iteration while an iteration of Word is roughly equivalent to two Gibbs iterations. Since one typically runs thousands of Gibbs iterations (while ours runs in 10 iterations even on very large data sets, yielding a running time equivalent to approximately 30 Gibbs iterations), we can observe several orders of magnitude improvement in speed by our algorithm. Further, running time could be significantly enhanced by noting that the Word algorithm trivially parallellizes.

4.2 Real Documents

We consider two real-world data sets with different properties: a random subset of the Enron emails (8K documents,

[2]The ℓ_1 distance used in (Podosinnikova *et al.*, 2015) is the normalized version, which is half of what we report here.

#Docs	SynthA					SynthB				
	2K	4K	6K	8K	10K	2K	4K	6K	8K	10K
KMeans	0.794	0.801	0.864	0.635	0.714	1.063	1.048	1.022	0.921	0.952
Basic	1.821	1.816	1.823	1.814	1.818	1.804	1.798	1.805	1.796	1.794
Word	0.772	0.384	0.373	0.364	0.220	0.944	0.760	0.582	0.537	0.504
VB	0.283	0.247	0.305	0.117	0.059	0.500	0.521	0.448	0.443	0.392
Spectral	0.310	0.169	0.158	0.149	0.112	0.494	0.384	0.372	0.296	0.314
JD	0.208	0.151	0.126	0.111	0.099	0.310	0.238	0.199	0.178	0.161
TPM	0.206	0.149	0.125	0.110	0.099	0.303	0.230	0.193	0.170	0.153
Anchor	0.179	0.142	0.120	0.107	0.102	0.144	0.135	0.118	0.118	0.112
W+R	0.141	0.107	0.093	0.086	0.080	0.102	0.131	0.155	0.110	0.105
CGS	0.130	0.092	0.076	0.199	0.197	0.094	0.091	0.098	0.338	0.276

Table 2: Comparison of topic reconstruction errors of different algorithms.

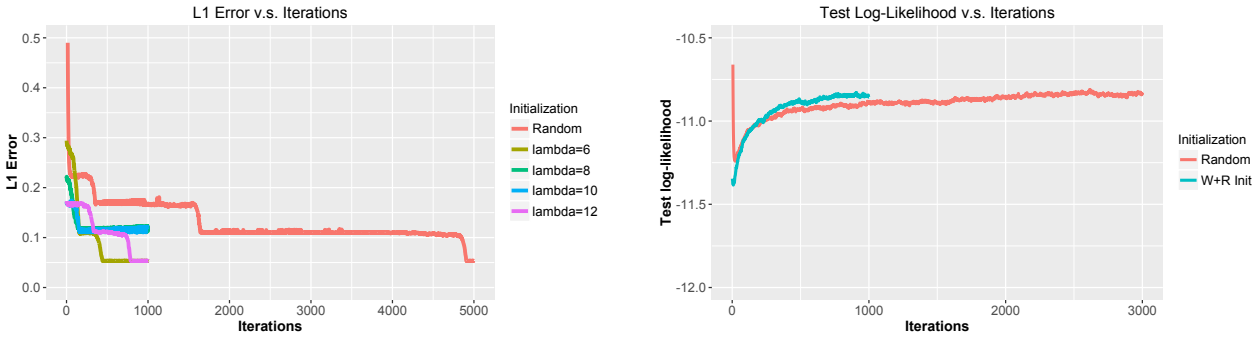

Figure 1: **left**: The evolution of topic reconstruction ℓ_1 errors of Gibbs sampler with different initializations: "Random" means random initialization, and "lambda=6" means initializing with the assignment learnt using `Word+Refine` algorithm with $\lambda = 6$ for 3 iterations. **right**: The evolution of the held-out word log-likelihood of Gibbs sampler with different initializations for the Enron dataset (best viewed in color).

	semiEnron	semiNYTimes
Basic	1.881	1.890
Word	0.529	0.721
VB	0.375	0.468
Spectral	0.340	0.510
JD	0.219	0.325
TPM	0.215	0.328
Anchor	0.199	0.313
W+R	0.201	0.297
CGS	0.202	0.283

Table 3: Comparison of topic reconstruction errors of different algorithms on the semi-synthetic datasets.

vocabulary size 5000), and a subset of the New York Times articles[3] (15K documents, vocabulary size 7000). 1K documents are reserved for predictive performance assessment for both datasets.

Semi-synthetic corpora. Following (Arora *et al.*, 2013), we generate semi-synthetic corpora from models trained with $K = 50$ from Enron and NYTimes for 3000 Gibbs steps, with document lengths set to 200. This setup gives a clear expected advantage to the performance of the Gibbs sampler; the main interest here is in comparisons to other methods. Table 3 shows the mean topic reconstruction errors for 10K documents and Figure 2 presents the density plot of the reconstruction errors. Similar to the synthetic situations, the Gibbs sampler has the lowest ℓ_1 error, where `Word+Refine` and `Anchor` come next. However, `Word+Refine` has the smallest error range. The `Word+Refine` also improves significantly over the `Basic` and `Word` algorithms. Again, the true parameters are provided as input when applicable.

Predictive performance. We consider the held-out word log-likelihood for predictive performance: fifty percent of the words of the test documents are used for inference and the other fifty percent forms the prediction set, which is similar to the document completion evaluation metric in (Wallach *et al.*, 2009)[4]. In addition, we also consider a *dis-*

[3] http://archive.ics.uci.edu/ml/machine-learning-databases/bag-of-words/

[4] Computing directly from the predictive distribution requires computationally demanding sampling procedures. As pointed out in (Ranganath *et al.*, 2015), it only allows testing of a smaller number (50) of documents.

Enron	$\beta = 0.1$		$\beta = 0.01$		$\beta = 0.001$	
	discrete	held-out	discrete	held-out	discrete	held-out
CGS	-5.932	-8.604	-5.484	-10.870	-5.091	-13.484
W+R	-5.434	-9.883	-5.147	-11.753	-4.918	-13.882
NYT	$\beta = 0.1$		$\beta = 0.01$		$\beta = 0.001$	
	discrete	held-out	discrete	held-out	discrete	held-out
CGS	-6.594	-9.345	-6.205	-11.431	-5.891	-13.813
W+R	-6.105	-10.666	-5.941	-12.296	-5.633	-14.639

Table 4: The held-out word log-likelihood on new documents for Enron ($K = 100$ topics) and NYTimes ($K = 100$ topics) datasets with fixed α value. See text for details.

CGS	art, artist, painting, museum, century, show, collection, history, french, exhibition
W+R	painting, exhibition, portrait, drawing, object, photograph, gallery, flag, artist
CGS	plane, flight, airport, passenger, pilot, aircraft, crew, planes, air, jet
W+R	flight, plane, passenger, airport, pilot, airline, aircraft, jet, planes, airlines
CGS	car, driver, truck, vehicles, vehicle, zzz_ford, seat, wheel, driving, drive
W+R	car, driver, vehicles, vehicle, truck, wheel, fuel, engine, drive, zzz_ford

Table 5: Example topics pairs learned from NYTimes dataset.

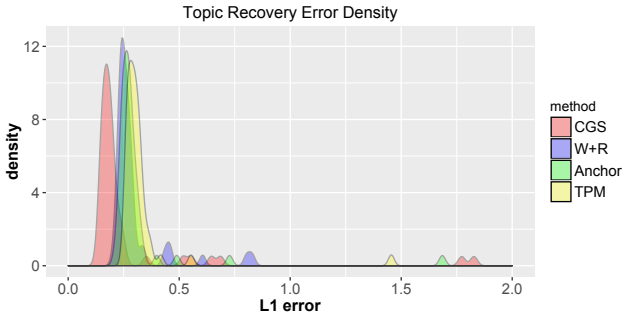

Figure 2: Density plot of the topic recovery error on the semi-NYTimes corpora for different algorithms (best viewed in color).

crete version of the test log-likelihood similar to k-means, where the computation is based on the learned topic and the word-topic assignment inferred by the Word algorithm. Table 4 shows the results on the Enron and NYTimes datasets. We can see that our approach excels in the discrete test log-likelihood while lags in the held-out log-likelihood. Note here we tune the λ value such that the resulting number of topics per document is comparable to that of the sampler. This would put our method at a disadvantage since the differences among the weights will be much smaller in the document-topic distribution. With a higher λ value, we can get comparable predictive performance as the sampler. For $\beta = 0.01$, the held-out log-likelihood of our method comes at -10.784 (v.s. -10.870) for Enron and -11.449 (v.s. -11.431) for NYTimes. In Figure 1, we also show the evolution of held-out word log-likelihood initialized with the Word+Refine optimized assignment for only 3 iterations. With these semi-optimized initializations, we again observe quite a speed-up over the random initialization. Table 5 further shows some sample topics generated by CGS and our method (see the supplement for the full list).

5 Conclusions

Our goal has been to lay the groundwork for a combinatorial optimization view of topic modeling as an alternative to the standard probabilistic framework. Small-variance asymptotics provides a natural way to obtain an underlying objective function, using the k-means connection to Gaussian mixtures as an analogy. We saw that the basic batch algorithm, as often utilized by researchers of small-variance techniques, performs poorly when compared quantitatively to probabilistic approaches. However, using ideas from facility location and incremental refinement, we obtained an algorithm that compares favorably, while being efficient and robust to initializations and parameter selection. Moreover, we also showed that the sampler's mixing time improves substantially when initialized using our method. Potential future work includes distributed implementations for further scalability, adapting k-means-based semi-supervised clustering techniques to this setting, and extensions of k-means++ (Arthur and Vassilvitskii, 2007) to derive explicit performance bounds for this problem.

Acknowledgement

This research was supported in part by NSF CAREER Award 1559558.

References

A. Anandkumar, Y. Liu, D. J. Hsu, D. P. Foster, and S. M. Kakade. A spectral algorithm for latent Dirichlet allocation. In *NIPS*, pages 917–925, 2012.

S. Arora, R. Ge, and A. Moitra. Learning topic models–

going beyond SVD. In *Foundations of Computer Science (FOCS)*, pages 1–10. IEEE, 2012.

S. Arora, R. Ge, Y. Halpern, D. Mimno, A. Moitra, D. Sontag, Y. Wu, and M. Zhu. A practical algorithm for topic modeling with provable guarantees. In *ICML*, 2013.

D. Arthur and S. Vassilvitskii. k-means++: The advantages of careful seeding. In *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2007.

A. Banerjee, S. Merugu, I. S. Dhillon, and J. Ghosh. Clustering with Bregman divergences. *Journal of Machine Learning Research*, 6:1705–1749, 2005.

T. Bansal, C. Bhattacharyya, and R. Kannan. A provable SVD-based algorithm for learning topics in dominant admixture corpus. In *NIPS*, pages 1997–2005, 2014.

D. P. Bertsekas. *Nonlinear Programming*. Athena Scientific, Belmont, Massaschusetts, 1999.

D. M. Blei and J. D. Lafferty. Correlated topic models. In *NIPS*, 2006.

D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3(4-5):993–1022, 2003.

D. M. Blei, M. I. Jordan, T. L. Griffiths, and J. B. Tenenbaum. Hierarchical topic models and the nested Chinese restaurant process. In *NIPS*, 2004.

T. Broderick, B. Kulis, and M. I. Jordan. MAD-Bayes: MAP-based asymptotic derivations from Bayes. In *ICML*, 2013.

T. Campbell, M. Liu, B. Kulis, J. How, and L. Carin. Dynamic clustering via asymptotics of the dependent Dirichlet process. In *NIPS*, 2013.

S. Deerwester, S. Dumais, T. Landauer, G. Furnas, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.

I. S. Dhillon, Y. Guan, and J. Kogan. Iterative clustering of high dimensioanl text data augmented by local search. In *IEEE International Conference on Data Mining (ICDM)*, 2002.

T. L. Griffiths and M. Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101:5228–5235, 2004.

T. Hofmann. Probabilistic latent semantic indexing. In *Proc. SIGIR*, 1999.

J. H. Huggins, K. Narasimhan, A. Saeedi, and V. K. Mansinghka. JUMP-means: Small-variance asymptotics for Markov jump processes. In *ICML*, 2015.

K. Jain, M. Mahdian, E. Markakis, A. Saberi, and V. V. Vazirani. Greedy facility location algorithms analyzed using dual fitting with factor-revealing LP. *Journal of the ACM*, 50(6):795–824, 2003.

K. Jiang, B. Kulis, and M. I. Jordan. Small-variance asymptotics for exponential family Dirichlet process mixture models. In *NIPS*, 2012.

H. W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955.

B. Kulis and M. I. Jordan. Revisiting k-means: New algorithms via Bayesian nonparametrics. In *ICML*, 2012.

J. Lee and S. Choi. Bayesian hierarchical clustering with exponential family: Small-variance asymptotics and reducibility. In *Artificial Intelligence and Statistics (AISTATS)*, 2015.

A. Q. Li, A. Ahmed, S. Ravi, and A. J. Smola. Reducing the sampling complexity of topic models. In *ACM SIGKDD*, pages 891–900. ACM, 2014.

A. Podosinnikova, F. Bach, and S. Lacoste-Julien. Rethinking LDA: moment matching for discrete ica. In *NIPS*, pages 514–522, 2015.

R. Ranganath, L. Tang, L. Charlin, and D. M. Blei. Deep exponential families. In *AISTATS*, 2015.

S. Roweis. EM algorithms for PCA and SPCA. In *NIPS*, 1997.

A. Roychowdhury, K. Jiang, and B. Kulis. Small-variance asymptotics for hidden Markov models. In *NIPS*, 2013.

R. Samdani, K-W. Chang, and D. Roth. A discriminative latent variable model for online clustering. In *ICML*, 2014.

Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. Hierarchical Dirichlet processes. *Journal of the American Statistical Association (JASA)*, 101(476):1566–1581, 2006.

Y. W. Teh, D. Newman, and M. Welling. A collapsed variational Bayesian inference algorithm for latent Dirichlet allocation. In *NIPS*, 2006.

S. Tong and D. Koller. Restricted Bayes optimal classifiers. In *AAAI*, 2000.

H. Wallach, I. Murray, R. Salakhutdinov, and D. Mimno. Evaluation methods for topic models. In *ICML*, 2009.

X. Wang and E. Grimson. Spatial latent Dirichlet allocation. In *NIPS*, 2007.

Y. Wang and J. Zhu. Small-variance asymptotics for Dirichlet process mixtures of SVMs. In *Proc. Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

I. E. H. Yen, X. Lin, K. Zhang, P. Ravikumar, and I. S. Dhillon. A convex exemplar-based approach to MAD-Bayes Dirichlet process mixture models. In *ICML*, 2015.