On-the-Fly Rectification for Robust Large-Vocabulary Topic Inference (Supplementary Material)

Anonymous Authors[1]

Abstract

Our supplementary material aims to help readers better understand the main draft. It answers to a wide range of potential questions from motivation of research to experimental details. It not only consists of the theoretical proofs and empirical supports for our lemmas and findings, but also presents additional result panels that are missing from the main paper due to space limit. We hope our readers find complementary guidance from reading this material.

1. Motivations and Contributions

Q1: What are the major problems of using co-occurrence information?
A1: When data consists of collections of discrete objects, using co-occurrence provides statistical semantics to individual objects (e.g., words, products, items) based on their relations with other objects through the data. As a result, co-occurrence statistics has been widely used for various unsupervised learning, but there are two major drawbacks:

- Individual objects naturally exhibit different burstiness. *Frequent objects* that too often co-occurs with many other objects would not provide meaningful semantics. Typical examples in textual data are stop words whose functions are limited to building syntax. *Rare objects* are noisy and unstable because these objects barely appear in the data, making co-occurrences with them prone to poor estimation. But they often provide exclusive markers (like topics) in underlying geometry of the data. Proper tuning of frequency imbalance is the key to successful learning, but it relies mostly on manual elaboration and domain-specific knowledge without any principled approach.

- Co-occurrence provides a great scalability by summarizing millions and billions of examples (e.g., documents, purchases, playlists). However, even the lightest format: co-occurrence matrix between pairs of objects, grows quadratically in the size of vocabulary. Tensor-based models often require co-occurrence between triples of objects, quickly exceeding normal computational budget as the size of vocabulary grows. Improving time and space complexities are crucial for usability of spectral posterior inference for latent variable models.

Q2: Why large vocabulary matters?
A2: Capability to process large vocabulary is the main challenge of the field in the following two reasons:

- Users in representation learning are often required to learn proper data representations for all or majority of their objects. Products in online shopping or items like movies and songs in streaming services form long-tail markets. Thus learning low-dimensional representations only for the frequently appearing objects is not adequate for their profit structures.

- Topic modeling based on co-occurrence often associates each topic with an *anchor word*, which dedicates only to the topic without (or weakly) contributing to the other topics. Using large vocabulary allows users to find better anchor words. In other words, most topic models with large vocabularies are proven to satisfy *separability assumption* (Ding et al., 2015). Our submission shows that topic quality improves both quantitatively and qualitatively as we increase the size of vocabulary.

- Using a large vocabulary improves less subjective interpretation of topics. Our model is capable of finding more specific characteristic words (i.e., relatively rare but associated tightly to each topic), which provide complementary information for interpreting individual topics aligning with their prominent words (i.e., mostly frequent words that are easy to understand).

Q3: Why matrix factorization? What makes rectification interesting?
A3: Matrix/Tensor factorization is a general notion to represent a variety of machine learning problems. Once writing the target inference problem as a factorization form, one

[1] Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

can identify some necessary geometries with clarity. For example, low-rankness is often an universal assumption for unsupervised learning of finding a compact underlying geometry.

- When we have a full generative model of the input data, we can construct co-occurrence as an unbiased moment estimator of the generative process. For instance, if the data exactly follows popular probabilistic processes of topic modeling, the co-occurrence matrix of words must be entry-wise non-negative ($\mathcal{NN}$), normalized to sum to one ($\mathcal{NOR}$), and positive semi-definite ($\mathcal{PSD}$) in addition to be low-rank (Lee et al., 2015). However, an empirical co-occurrence cannot jointly satisfy these structures due to statistical noises from the finite samples. **Rectification** provides a principled treatment to fix statistical unstability by projecting the empirical co-occurrence (constructed from the data) onto a manifold consistent with these necessary geometries, providing an innovative way to improve posterior inference.

- It turns out that major spectral algorithms for latent variable generative models suffer from the similar issue, so called **model-data mismatch** (Kulesza et al., 2014). Rectification could open a new solution to fix the issue prevalent in spectral inference of mixed-membership latent variable models. In addition, performance of word-vector embeddings depends highly on how to correct co-occurrences with rare and frequent objects (Levy et al., 2015; Pennington et al., 2014). The models themselves do not provide any guidance to fix such co-occurrences, requiring labor-intensive tuning for successful learning of vectorial representations. As their learning tasks can also be written as forms of matrix factorization (Levy & Goldberg, 2014), rectification could be applicable to low-dimensional embedding learning without any ad-hoc treatment.

2. Foundations and Comparisons to Previous Work

Q4: It is not easy to understand connections between probabilistic and spectral topic modeling.
A4: Our submission tries not to use probability notations. It helps increasing consistency but decreases readability. Among many explanations to describe foundations in probabilistic and spectral topic models, we find Section 2 in (Lee et al., 2020) is particularly insightful. Here we retype it for your convenience.

We begin this section with a formal introduction to spectral topic modeling. Consider a dataset of M documents consisting of tokens drawn from a vocabulary of N words. Topic models assume that K topics are used to generate this dataset, where each topic is a distribution over the words; we summarize the latent topics by the column-stochastic matrix $\boldsymbol{B} \in \mathbb{R}^{N \times K}$ where each column $\boldsymbol{b}_k \in \Delta^{N-1}$ represents the distribution of the topic k. For each document m, choose a topic composition $\boldsymbol{w}_m \in \Delta^{K-1}$ first from a certain prior $\mathfrak{f}$; we collect these hidden compositions into another column-stochastic matrix $\boldsymbol{W} \in \mathbb{R}^{K \times M}$. These models assume that each of the n_m tokens in the document m is then generated independently from the categorical distribution given by the word-probability vector $\boldsymbol{B}\boldsymbol{w}_m \in \mathbb{R}^N$.

Different models adopt different $\mathfrak{f}$ such as $\mathfrak{f} = \text{Dir}(\boldsymbol{\alpha})$ for Latent Dirichlet Allocation (LDA) (Blei et al., 2003); $\mathfrak{f} = \mathcal{LN}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ or $\mathfrak{f} = \mathcal{PN}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ for Logistic/Probit-Normal Correlated Topic Models (CTMs) (Blei & Lafferty, 2007; Yu & Fokoue, 2014). Let $\boldsymbol{H} \in \mathbb{R}^{N \times M}$ be the word-document matrix where the m-th column vector $\boldsymbol{h}_m$ counts the occurrences of each word in the document m, and let $\widetilde{\boldsymbol{H}}$ be the column-normalized version of $\boldsymbol{H}$ that specifies the relative frequencies of each word rather than the raw counts. Then topic modeling aims to learn *latent topics* $\boldsymbol{B}$ and *hidden compositions* $\boldsymbol{W}$ given the *observed collections of words* $\boldsymbol{H}$. Equivalently, we seek a non-negative matrix factorization $\widetilde{\boldsymbol{H}} \approx \boldsymbol{B}\boldsymbol{W}$ but with a prior to make individual topic compositions $\{\boldsymbol{w}_m\}$ coherent within a corpus.

Joint Stochastic Matrix Factorization (JSMF) The matrix $\widetilde{\boldsymbol{H}}$ of word frequencies is sparse, noisy, and often inconveniently large. Let us consider instead the word co-occurrence matrix $\boldsymbol{C} \in \mathbb{R}^{N \times N}$, where $\boldsymbol{C}_{ij}$ indicates the joint probability of observing a pair of words (i, j). Then topic modeling corresponds to a second-order non-negative matrix factorization: $\boldsymbol{C} \approx \boldsymbol{B}\boldsymbol{A}\boldsymbol{B}^{\mathsf{T}}$ where the column-stochastic matrix $\boldsymbol{B} \in \mathbb{R}^{N \times K}$ represents the topics as before and the joint-stochastic matrix $\boldsymbol{A} \in \mathbb{R}^{K \times K}$ represents the *topic correlations*. If the true compositions $\boldsymbol{W}^*$ that generate the data are known, we can define the true correlations by $\boldsymbol{A}^* := \frac{1}{M}\boldsymbol{W}^*\boldsymbol{W}^{*\mathsf{T}}$ where $\boldsymbol{A}^*_{kl}$ is the joint probability for a pair of latent topics (k, l). By forming $\boldsymbol{C}$ as an unbiased estimator of the underlying generative process, we can identify $\boldsymbol{B}$ and $\boldsymbol{A}$ close to the true topics and their correlations.[1]

It is helpful to compare the matrix-based view of JSMF to the generative view of standard topic models. For each document m, the generative view begins with the topic composition $\boldsymbol{w}_m$, focusing on how to produce streams of tokens. We keep choosing a topic z from $\boldsymbol{w}_m$ and then a word x from $\boldsymbol{b}_z$ for each of the n_m positions. The correlations between words that $\boldsymbol{w}_m \sim \mathfrak{f}$ induces are not explicitly modeled. In contrast, the matrix-based view starts with *individual topic correlations* $\boldsymbol{A}_m$ for each document m. Then for each of the possible $n_m(n_m - 1)$ position pairs, *a pair of topics* (z_1, z_2) is selected first from $\boldsymbol{A}_m$, then *a pair of words* (x_1, x_2) is

[1] As the number of documents M grows, $\boldsymbol{A}$ converges to the true $\boldsymbol{A}^*$ and the prior $\mathbb{E}_{\boldsymbol{w}\sim\mathfrak{f}}[\boldsymbol{w}\boldsymbol{w}^T]$ (Arora et al., 2012).

chosen according to the topics $(\boldsymbol{b}_{z_1}, \boldsymbol{b}_{z_2})$, respectively. The word co-occurrence matrix explicitly captures the resulting correlations induced by the *prior topic correlations* $\boldsymbol{A}$. This pair generation view has the following two important implications: Recall that sharing the prior $\mathfrak{f}$ for $\{\boldsymbol{w}_m\} \sim \mathfrak{f}$ is the crux of modern topic modeling (Asuncion et al., 2009), and our flexible matrix prior $\boldsymbol{A}$ takes the role of $\mathfrak{f}$ for JSMF.

Q5: Why don't you compare against existing methods like Variational Inference, Gibbs Sampling or tensor-based spectral topic models? Why only comparing to AP for rectification?
A5: We mainly compare against RAW (=AP+AW) as our main contribution is to scale the anchor-based algorithms to larger vocabularies. Earlier work (Lee et al., 2015) shows that RAW learns quality topics comparable to those from Collapsed Gibbs Sampling. Previous work (Lee et al., 2019) already shows that RAW performs both quantitatively and qualitatively better than Online Variational Inference and the tensor-based spectral method with CP-decomposition. It also tests Douglas-Rachford rectification (DR) instead of AP for RAW, but there was no difference in performance. Therefore our submission compares the performance only to RAW.

Q6: Likelihood-based methods do not suffer from the model-data mismatch issue. What are the benefits of using spectral methods instead of using standard probabilistic inference?
A6: Whereas likelihood-based methods keeps iterating through the input data (e.g., documents or playlists) until convergence, spectral methods no longer revisit the input data once it is summarized into a co-occurrence statistics. Note that we often operate on millions or billions of documents, whereas the size of vocabulary is order-of-magnitudes smaller. In addition, spectral methods are

- Consistent. Predictions are consistent in the limit of infinite data.
- Transparent. The overall inference consists of clearly separated three steps: co-occurrence construction, rectification, and factorization.
- Interpretable. Each topic associates with a unique identifier called anchor words. Our Table 1 show that the *Characteristic Words (CWs)* from each anchor word can contribute to a specific understanding of each topic.
- Provable. Though this is not the main point of our paper, one can show finite sample convergence bounds.

Q7: For large vocabularies, the original Anchor Word algorithm already uses random projections of co-occurrence given Johnson-Lindenstrauss lemma. What makes your work different?
A7: The original Anchor Word algorithm (AW) (Arora et al., 2013) uses two types of random projections. Both transform the empirical co-occurrence into low-dimensional spaces. However,

- As studied in (Lee & Mimno, 2014), the projected co-occurrence matrix has many non-negligible negative entries, which break the next step of inferring topics $\boldsymbol{B}$ (or concretely topic-word matrix $\breve{\boldsymbol{B}}$) as this step is based mainly on Bayes Rule. As an additional result, the learned topic correlations $\boldsymbol{A}$ consist of many negative entries, being far from a legal joint distribution.
- Even if the random projections are used only for fast finding of anchor words, the result set of anchors $\boldsymbol{S}$ cannot be better than the anchor words found in the original space. Our Figure 1 clearly demonstrates that the anchor words found from the raw co-occurrence without any rectification are far from covering the convex hull of co-occurrence space.

The key implication is that *rectification* is an essential step for high-quality topic and correlation inference as well as high-quality anchor finding. But the previously available rectifications like AP and DR cannot scale to increasing vocabularies. Our contribution: ENN and PALM are to simultaneously achieve compression and rectification of co-occurrence statistics. Our another contribution LAW indeed scales similarly to AW with random projection, but it infers topics and topic correlations more naturally from the outputs of ENN and PALM. In particular, it avoids forming any rectified $\boldsymbol{C}$, which would incur a cost of $\mathcal{O}(N^2)$. In our LR-JSMF pipeline, we can completely bypass this $\mathcal{O}(N^2)$ constraint by constructing a low-rank approximation of the full co-occurrence directly from the raw data.

3. Algorithms and Pipeline

Q8: The paper has a number of different but related algorithms. Which algorithm is used in which step? What algorithms are from prior work and what are new in this work?
A8: Our framework consists of three main steps: 1) co-occurrence construction; 2) rectification; and 3) factorization. See our flowchart in Figure 1 together with the following explanations.

- For rectification, Algorithm 2 (RAW) consists of Alternating Projection rectification (AP) from the prior work (Lee et al., 2015). For simultaneous rectification and compression, we propose the *Epsilon Non-negative rectification (ENN)* in Algorithm 3 and the *Proximal Alternating Linearized Minimization rectification (PALM)* in Algorithm

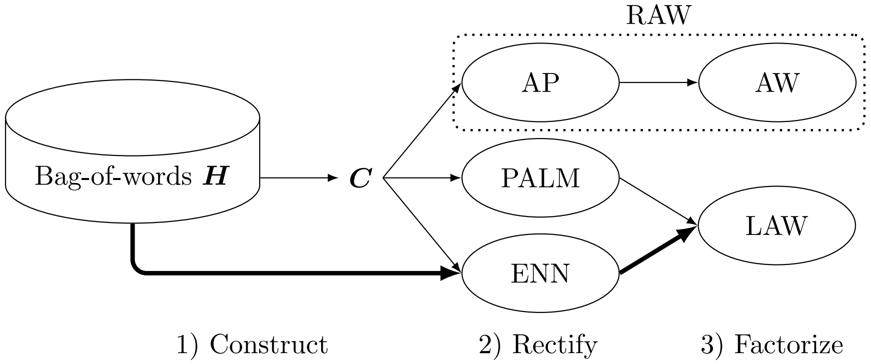


Figure 1. Flowchart of the framework. The bold arrows represent our proposed pipeline, LR-JSMF.

4. Note that AP outputs $N \times N$ rectified co-occurrence $\boldsymbol{C}^*$, whereas our ENN and PALM output $N \times K$ rectified compression of co-occurrence $\boldsymbol{Y}$.

- For inference given full co-occurrence $\boldsymbol{C}$, we use the original Anchor Word algorithm (AW) in Algorithm 1 from the prior work (Arora et al., 2013). For inference given the rectified and compressed co-occurrence $\boldsymbol{Y}$, we propose the *Low-rank Anchor Word algorithm (LAW)*. AP+AW, together known as the Rectified Anchor Word algorithm (RAW), is from the previous work (Lee et al., 2015).

- From co-occurrence construction to factorization, our *Low-Rank Joint Stochastic Matrix Factorization (LR-JSMF)* constructs only a low-rank approximation of the input data $\boldsymbol{H}$, bypassing the construction of any full co-occurrence $\boldsymbol{C}$. Then it enjoys total computational cost linear in N by initializing ENN by the low-rank approximation of the input data.

Q9: ENN could lead to a co-occurrence matrix with negative values. Is it fine?
A9: ENN only leads to *tiny* negative values in the decompressed version $\boldsymbol{C}^* = \boldsymbol{Y}\boldsymbol{Y}^\mathsf{T}$. Our experimental results show that ENN+LAW has no visible difference from AP+AW qualitatively in the learned topics as well as quantitatively in a number of metrics. Indeed, strictly enforcing non-negativity constraint of the decompressed $\boldsymbol{C}^*$ in PALM deteriorates the quality of topics although they still perform much better than the original Anchor Word algorithm (AW) without any rectification. Note also that many other previous already claim provable guarantees under even bigger violations in non-negativity (Arora et al., 2012; 2013; 2016). Different from these papers, our ENN can also control the degree of violation ϵ.

Q10: What is the intuition behind using column-pivoted QR on $\overline{\boldsymbol{C}}^\mathsf{T}$ to find the anchor words?
A10: Previous work (Lee et al., 2015) shows that separability enables us to represent all non-anchor rows in $\overline{\boldsymbol{C}}$ as a convex combination of the anchor rows (i.e. $\overline{\boldsymbol{C}}_{i*} = \sum_k \breve{\boldsymbol{B}}_{ki}\overline{\boldsymbol{C}}_{s_k*}$). To infer topics (concretely topic-word matrix $\breve{\boldsymbol{B}}$), therefore, we need to find K number of rows in $\overline{\boldsymbol{C}}$ that form a large enough convex hull to enclose all the other rows, the co-occurrence space in Figure 1. *Column-pivoted QR* is a greedy approach to this problem: in each step, we pick a pivot that is farthest away from the points picked so far, orthogonally project all the remaining points, and repeat this Gram-Schmidt process until we pick K points. We particularly use the algorithm in (Lee et al., 2019) for finding the set of anchor words $\boldsymbol{S}$.

Q11: The Low-rank Anchor Word algorithm (LAW) shown in Algorithm 5 involves computing the QR decomposition of $\boldsymbol{Y} = \boldsymbol{Q}\boldsymbol{R}$. What is the additional cost incurred by this step?
A11: Computing the QR decomposition of $\boldsymbol{Y} \in \mathbb{R}^{N \times K}$ takes $\mathcal{O}(NK^2)$ time without affecting to our computational complexity linear in N.

4. Dataset and Experimental Results

Q12: How exactly is the raw data pre-processed before constructing co-occurrence matrix $\boldsymbol{C}$?
A12: Given the bag-of-words, we first filter out all stop words such as $\{a, \textit{the}, \textit{of}, \dots\}$. Then we measure the tf-idf scoring of all words and keep the top N words, where N is the user-specified vocabulary size. Unlike the usual tf-idf scoring, we integer floor all inverse document frequency scores. This consequently removes words that occur in more than 50% of the documents. Lastly, we discard all documents that contain less than 3 unique words or are less than 5 tokens long. Note that this produces exactly identical dataset on which the previous work measures the performance of RAW (=AP+AW) (Lee et al., 2015; 2019; 2020).

Q13: For most datasets used, the number of documents M exceeds the vocabulary size N. In such cases, does $\mathcal{O}(NMK)$ runtime of the proposed LR-JSMF pipeline exceed $\mathcal{O}(N^2K)$?

A13: Since the word-document matrix $\boldsymbol{H}$ is much sparser than the word co-occurrence $\boldsymbol{C}$, using $\boldsymbol{H}$ to construct compressed co-occurrence statistics $\boldsymbol{V}$ and $\boldsymbol{D}$ turns out to be much more efficient than explicitly constructing $\boldsymbol{C}$ even in cases where $M > N$. To be specific, a tighter bound on the cost of initializing ENN directly from $\boldsymbol{H}$ gives $\mathcal{O}(LMK)$ time and $\mathcal{O}(NK)$ space, where L is the average number of unique words in each document. For the previous Rectified Anchor Word algorithm (RAW), the tighter bounds of constructing $\boldsymbol{C}$ are $\mathcal{O}(L^2M)$ time and $\mathcal{O}(N^2)$ space.

Q14: Why are the different metrics shown in Figures 2 and 3?
A14: We only present 7 metrics for each experiment for readability in the limited space. The panels that were missing are shown in Figure 2 of this supplementary. In addition to evaluation metrics discussed in the main paper, here we present two more metrics to maximize comparability with other work: **Sparsity** ($\frac{1}{K}\sum_k \frac{\sqrt{N}-(\|\boldsymbol{b}_k\|_1/\|\boldsymbol{b}_k\|_2)}{\sqrt{N}-1}$) measures how concentrated the topics are on specific words, and the traditional **Coherence** ($\frac{1}{K}\sum_k \sum_{x_1\neq x_2}^{x_1,x_2\in Top_k} \log \frac{D_2(x_1,x_2)+\epsilon}{D_1(x_2)}$) measures how often the top Prominent Words of each topic co-occur within training documents. $D_1(\cdot)$ and $D_2(\cdot,\cdot)$ denote document frequencies and co-document frequencies within the corpus, respectively.

Q15: For metrics involving $\boldsymbol{C}$, were they measured with the original $\boldsymbol{C}$ or the rectified $\boldsymbol{C}$?
A15: All metrics presented are measured against the original $\boldsymbol{C}$. Particularly for the large-vocabulary experiments in Figure 3, we use approximated versions of RelativeRecovery and RelativeApproximation instead since vocabulary sizes over 15k prohibit explicit use of full co-occurrence $\boldsymbol{C}$ on standard hardware. We verified that the approximated versions agree with the exact versions in their trends when measuring on the datasets with small vocabularies.

Q16: Why the patterns over increasing x-axis values are not always consistent?
A16: Say we generate a dataset from a probabilistic topic model with two topics. As topic modeling is also a clustering, learning two or four clusters from the dataset is much easier than learning three clusters or five clusters. Therefore our metric values or runtimes are not necessarily monotone when increasing the number of topics. Similarly, the difficulty of learning some number of topics varies when changing the size of vocabularies, thereby creating some amount of inconsistency.

Q17: Why some of the metrics are changed to relative versions?
A17: We change three intrinsic metrics: Recovery error, Approximation error, and Diagonal Dominancy into their relative versions from the original ones in (Lee et al., 2015; 2017; 2019). This is because we here compare different sizes of vocabularies as well as different numbers of topics.

Q18: According to the source code, the proposed algorithms seem to use fixed number of iterations rather than running until convergence. Is it fair?
A18: It is primarily because the previous work (Lee et al., 2015) that we directly compare against uses the fixed number of iterations for its rectification step by AP. However, we significantly test both fixed number of iterations and fixed thresholds for convergence. Then we set the proper number of iterations for our ENN and PALM empirically, so that the learned topics are in convergence. However, such convergence in topics is not easy to be translated into the uniform convergence criteria for different rectifications. Although we cannot guarantee these number of iterations are optimal for every possible textual and non-textual datasets, our experimental designs are fair on these four datasets.

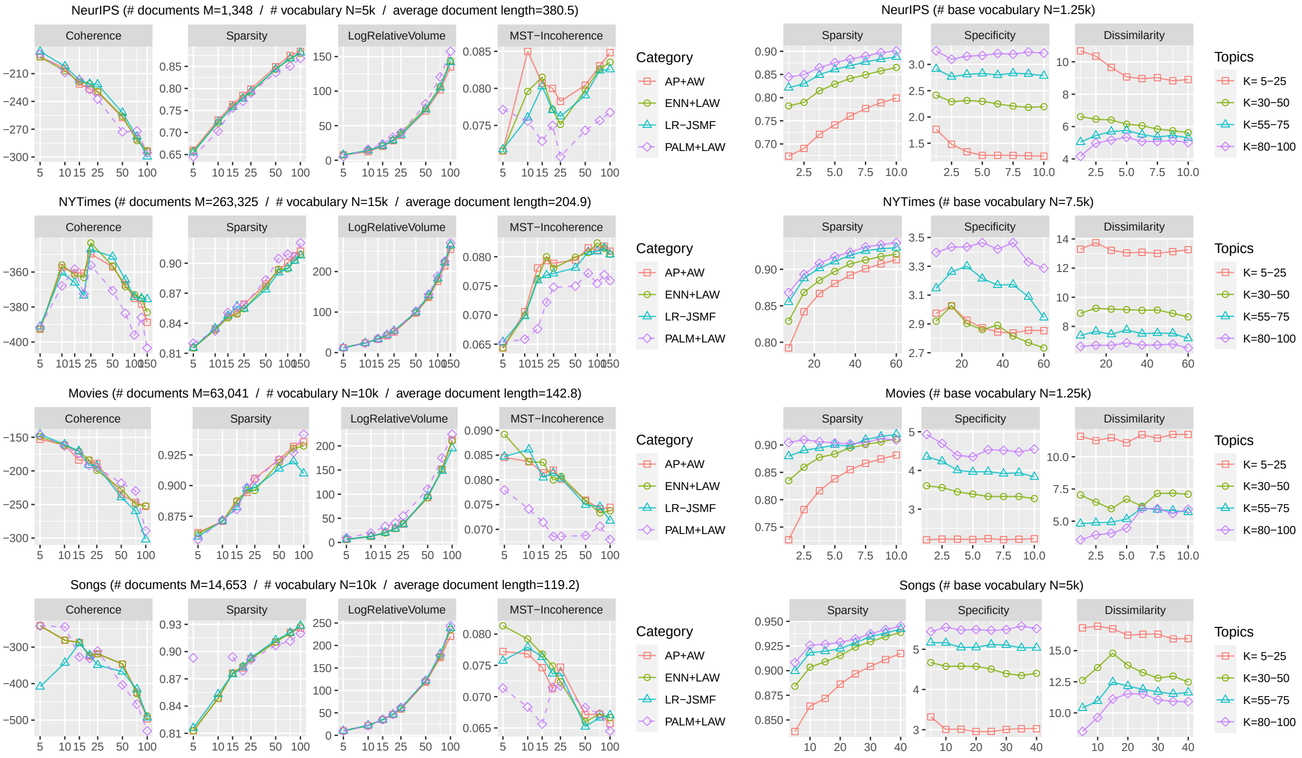
Q19: How did you choose the parameters in your algorithm?
A19: Most of them are selected empirically through careful experimentation. We are able to find these which perform well across all datasets we have tested on. Some other parameters in our numerical methods are by the recommendation of the original paper.

Q20: What was the hardware setup for the experiments presented in Section 5?
A20: All experiments were performed on a high-performance computing cluster running on RedHat 6.6 OS. The non-monotonic increase in runtimes are due to preemptive load-balancing of the cluster's job scheduler. To reduce fluctuation, we took the average over 10 runs for all runtime measurements shown in the main paper.

Q21: Will be source code be released upon publication?
A21: Yes. We will include a Github link to our libraries in the final draft. The anonymized source codes are submitted with the datasets used in the experiments.

5. Topic Variations when Increasing or Decreasing Vocabularies

Figure 3 shows qualitatively that users can learn more distinguishable topics by using larger vocabularies. It is especially because the algorithms make use of rarer words for more specific contexts. Moving from left to right columns, we can observe that the set of *Prominent Words (PWs)* becomes more specific. For instance, the topic corresponding to the third row are slightly vague when the sizes of vocabulries are smaller than $N\!=\!5000$, whereas we gain access to highly topic-specific words such as *hjb* (Hamilton-Jacobi-Bellman equation) or *pid* (Proportional Integral Derivative) when $N = 5000$, signifying the pertinence to dynamical and control systems of the topic. When using smaller vocabularies,



(a) Figure 2

(b) Figure 3

Figure 2. Additional panels for Figures 2 and 3 of the main paper. The x-axes show the number of topics K in Figure 2, and the vocabulary size N in Figure 3. In y-axes, higher is better for all metrics except for MST-Incoherence. We cannot measure Coherence for the increasing-vocabulary experiment in Figure 3 due to the large memory cost of storing co-document counts for every pair of words.

we also observe that words normally considered as less interesting terms can often contribute highly to topics. For example, the topic corresponding to the bottom row shows red shade when $N = 1250$, indicating that general terms such as *equivalent* or *cambridge* are strongly connected to the machine learning literature.

References

Arora, S., Ge, R., and Moitra, A. Learning topic models – going beyond SVD. In *FOCS*, 2012.

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

Arora, S., Ge, R., Koehler, F., Ma, T., and Moitra, A. Provable algorithms for inference in topic models. In *ICML*, pp. 2859–2867, 2016.

Asuncion, A., Welling, M., Smyth, P., and Teh, Y. W. On smoothing and inference for topic models. In *UAI*, 2009.

Blei, D. and Lafferty, J. A correlated topic model of science. *Annals of Applied Statistics*, 2007.

Blei, D., Ng, A., and Jordan, M. Latent Dirichlet allocation. *JMLR*, 2003.

Ding, W., Ishwar, P., and Saligrama, V. Most large topic models are approximately separable. In *ITA, 2015*, pp. 199–203. IEEE, 2015.

Kulesza, A., Rao, N. R., and Singh, S. Low-rank spectral learning. In *Artificial Intelligence and Statistics*, pp. 522–530, 2014.

Lee, M. and Mimno, D. Low-dimensional embeddings for interpretable anchor-based topic inference. In *EMNLP*. Association for Computational Linguistics, 2014.

Lee, M., Bindel, D., and Mimno, D. Robust spectral inference for joint stochastic matrix factorization. In *NIPS*, 2015.

Lee, M., Bindel, D., and Mimno, D. From correlation to hierarchy: Practical topic modeling via spectral inference. In *12th INFORMS Workshop on Data Mining and Decision Analytics*, 2017.

Lee, M., Cho, S., Bindel, D., and Mimno, D. Practical correlated topic modeling and analysis via the rectified anchor word algorithm. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 4992–5002, 2019.

Lee, M., Bindel, D., and Mimno, D. Prior-aware composition inference for spectral topic models. In *Artificial Intelligence and Statistics 2020 (AISTATS)*, 2020.

Levy, O. and Goldberg, Y. Neural word embedding as implicit matrix factorization. In *NIPS*, 2014.

Levy, O., Goldberg, Y., and Dagan, I. Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225, 2015.

Pennington, J., Socher, R., and Manning, C. D. GloVe: Global vectors for word representation. In *EMNLP*, 2014.

Yu, X. and Fokoue, E. Probit normal correlated topic model. In *Open Journal of Statistics*, pp. 879–888, 2014.

Vocabulary Size						
1250	2500	3750	5000 (base)	6250	7500	8750
refractory	interspike	bursting	neuron	signalling	ipsp	tst
interconnection	marder	stomatogastric	circuit	meilijson	stg	tam
seen	abbott	konishi	synaptic	quiescent	substances	hyperpolarized
detail	acad	axonal	cell	ryckebusch	inactivation	memorized
transmission	pyloric	modulatory	layer	leech	depolarized	transposed
considered	bird	ionic	signal	silent	shapiro	tsividis
san	male	kanji	recognition	radical	sicl	subword
additional	henderson	phonemic	layer	npm	joe	phoneroes
amount	jackel	subsystem	hidden	demi	chinese	otherfilter
considered	recog	dtw	word	shikano	hanazawa	sdnn
developed	dictionary	strokes	speech	letterform	lexicon	perplexity
significant	ocr	gender	net	tebelskis	preprocessed	males
cambridge	discounted	tutor	control	hjb	jacobi	ovi
pendulum	bradtke	lqr	action	rein	forcement	pid
requires	discount	disturbances	dynamic	biped	viscosity	idm
con	eligibility	disturbance	optimal	trol	sel	umass
bellman	indirect	hamilton	reinforcement	handicapped	bizzi	missile
plan	amherst	smdp	controller	gullapalli	swinging	queueing
directional	transparent	luminance	cell	unoriented	aftereffect	moc
seen	adelson	ruderman	field	heeger	blast	ori
dark	geniculate	andersen	visual	thalamus	mae	taube
neuroscience	amacrine	bergen	motion	directionally	knierim	muller
soc	deg	selectively	direction	hayashi	mexican	swindale
supported	mcnaughton	lond	image	werblin	specimen	skagg
equivalent	fix	solvable	gaussian	birmingham	boxplot	pbr
computing	satisfying	distribu	noise	kolmogorov	dependences	owi
cambridge	royal	barber	approximation	gmm	cb2	danish
considered	opper	parametrized	hidden	winther	trigonometric	ylz
simply	leibler	const	bound	imation	colt	diabetes
detail	treatment	eter	matrix	statist	minimizer	devroye

Figure 3. Losses or gains in Prominent Words (PWs) depending on the vocabulary size. Each row represents a topic from the NeurIPS dataset, with the top 6 PWs shown in the middle column. The red and green cells denote PWs that are lost or gained by shifting the vocabulary size from the default size $N = 5000$, respectively. The color intensities indicate the each word's contribution towards the corresponding topic.