Political Analysis (2026), **34**, 53–77
doi:10.1017/pan.2025.10024

PA

ARTICLE

Analyzing Political Text at Scale with Online Tensor LDA

Sara Kangaslahti[1], Danny Ebanks[2], Jean Kossaifi[3], Anqi Liu[4], R. Michael Alvarez[5] and Animashree Anandkumar[6]

[1]Department of Computer Science, Harvard University, USA; [2]Institute for Quantitative Social Science, NVIDIA Corp., USA; [3]Johns Hopkins University, USA; [4]Department of Computer Science, California Institute of Technology, USA; [5]Division of Humanities and Social Science, California Institute of Technology, USA; [6]Division of Engineering and Applied Sciences

Corresponding author: R. Michael Alvarez; Email: rma@caltech.edu

(Received 18 March 2024; revised 14 November 2024; accepted 22 November 2024; published online 4 December 2025)

Abstract

This article proposes a topic modeling method that scales linearly to billions of documents. We make three core contributions: i) we present a topic modeling method, tensor latent Dirichlet allocation, that has identifiable and recoverable parameter guarantees and sample complexity guarantees for large data; ii) we show that this method is computationally and memory efficient (achieving speeds over 3×–4× those of prior parallelized latent Dirichlet allocation methods), and that it scales linearly to text datasets with over a billion documents; and iii) we provide an open-source, GPU-based implementation of this method. This scaling enables previously prohibitive analyses, and we perform two real-world, large-scale new studies of interest to political scientists: we provide the first thorough analysis of the evolution of the #MeToo movement through the lens of over two years of Twitter conversation and a detailed study of social media conversations about election fraud in the 2020 presidential election. Thus, this method provides social scientists with the ability to study very large corpora at scale and to answer important theoretically-relevant questions about salient issues in near real-time.

Keywords: text data; tensor latent Dirichlet allocation; natural language processing; topic models

Edited by: Jeff Gill

1. Introduction

We propose a new method to estimate topic models that is feasible on large-scale data that has theoretical accuracy guarantees. Our approach leverages theoretical insights from Anandkumar *et al.* (2013) that show that a spectral decomposition approach to topic models scales to large datasets and possesses desirable theoretical properties, such as provable, accurate recovery of the parameters and large-sample consistency. To achieve scale, we show that by demeaning and batching the data, our method estimates topic model outputs for large-scale documents and recovers the same model as Anandkumar *et al.* (2013), endowing it with the same theoretical guarantees. This approach has many benefits for political scientists, who have used topic modeling methods to study important questions across the discipline, such as studies using text data to study new questions concerning political behavior (Barberá *et al.* 2015; Metzger *et al.* 2016; Munger 2017) and public opinion (Barberá 2015; Barberá *et al.* 2019). Text data has been important for new advances in analyzing the evolution of protest movements and social protests (Kann *et al.* 2023; Larson *et al.* 2019; Steinert-Threlkeld 2017; Tillery 2019). New methods for analyzing text data and more accessible data have allowed researchers to explore political communications, agenda setting, and the news media (Barberá *et al.* 2021; Gilardi *et al.* 2022).

In this article, we contribute to a rich line of methodological research in political science that has innovated and proposed clever frameworks to meet the needs of applied researchers across a wide

© The Author(s), 2025. Published by Cambridge University Press on behalf of The Society for Political Methodology.

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

variety of domains. From best practices and research design frameworks for how to incorporate text (Grimmer and Stewart 2013; Gilardi *et al.* 2022; King and Hopkins 2010), to approaches to unsupervised methods (Denny and Spirling 2018), political methodologists are guiding the field in how to best approach this high-dimensional data. Researchers have also introduced new tools for political scientists, including topic models which incorporate metadata (Roberts *et al.* 2014), computer-assisted techniques in processing texts in both clustering and in comparative settings (Grimmer and King 2011; Lucas *et al.* 2015), lexical feature selection (Monroe, Colaresi, and Quinn 2017), and crowd-sourcing approaches to measure sophistication (Benoit, Munger, and Spirling 2019). For example, in this article, we study two large-scale political science datasets. First, we study a dataset composed of tweets generated before and during the #MeToo movement to better understand the evolution of collective action and protest movements. The Women's March in January 2017 was the largest political protest in American history up until that time (Friedersdorf 2017). We also study a dataset of tweets generated after the 2020 Presidential election in order to better understand coordination effects, the loser's effect, and how online publics react to electoral defeat. These large, dynamic, and unstructured datasets offer new insights into mass politics and how it manifests on online discourse in particular. This is especially helpful where political scientists have been limited to surveys which rely on respondent recall and are generally static in nature.

Importantly, researchers are now collecting text datasets that are larger and larger in scale. For example, there are now numerous studies from different disciplines reporting the use of datasets that contain more than a billion tweets (Dimitrov *et al.* 2020; Hannak *et al.* 2021; Sinnenberg *et al.* 2016). However, for large datasets, typical approaches for the estimation of latent Dirichlet allocation (LDA) methods are often computationally impractical and memory inefficient. Data discovery and description techniques for these large data can help inform new theoretical frameworks, establish critical empirical facts, and help establish an empirical foundation for political science researchers to explore with rigorous tools from causal inference frameworks. In this article, we propose a new scalable online tensor LDA (TLDA) method with an end-to-end GPU implementation, ideally suited for the analysis of large text datasets. In summary, we make the following contributions:

- *LDA with Theoretical Foundations:* The method has *identifiable and recoverable* parameter guarantees and *sample complexity* guarantees for large data. These theoretical properties provide assurance that under the assumed data generation process and mild regularity assumptions, the method returns accurate results in large data.
- *Political Science Data Discovery:* Our method improves understanding of large corpora at scale and answers questions in real-time about politically salient topics, such as the #MeToo movement and social media activity around the Presidential election in 2020.
- *Fully Online, Incremental Tensor LDA:* Our method can be estimated in real-time without relying on a precomputed dimensionality reduction of the second-order moment. This results in a method that is computationally and memory efficient.
- *Scaling to Large Corpora:* We demonstrate that the method scales linearly by applying our approach to over 1 billion documents, scaling results which we show in the Supplementary Material.
- *Efficient Implementation with End-to-End GPU Acceleration:* In addition to our theoretical contributions, we release a new open-source library alongside this article, which provides an efficient GPU-based implementation of all steps of topic modeling from pre-processing to tensor operations, without costly GPU–CPU exchanges.[1]

To demonstrate the usefulness of our method, we utilize it to analyze the topical development of 8 million tweets concerning #MeToo from September 2017 through December 2019. Using our TLDA method, we can discern clear topical evolution over time through a qualitative study. Notably, as we dynamically grow the corpus of tweets, we find that topics related to politically salient news events are generally ephemeral. In contrast, the topical prominence related to personal testimonies, coordinating

[1] The package is available at https://tensorly.org/tlda/dev/.

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

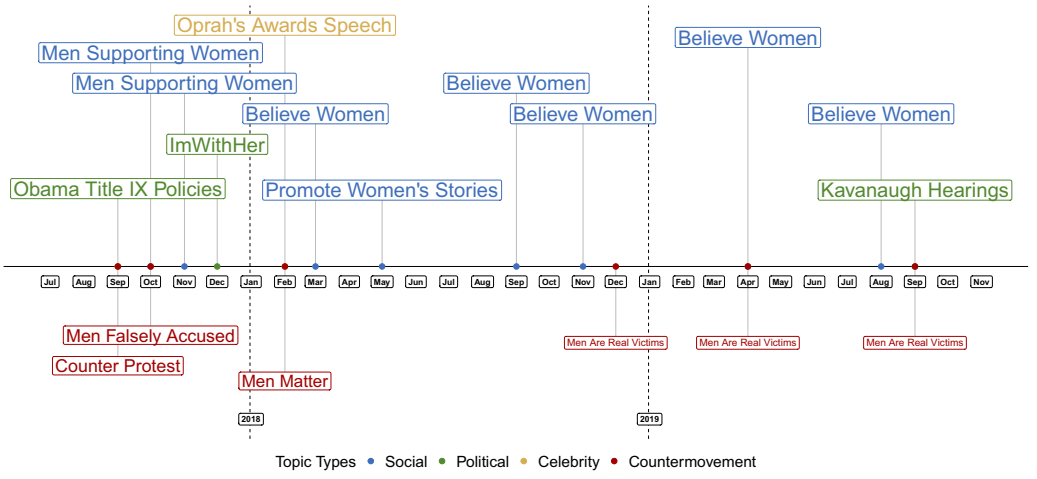


Figure 1. Evolution of the most prominent pro- and counter-movement topics in the #MeToo discussion.
Note: In each iteration of the dynamic analysis described in Section 7.2, we inspect the topics and manually label them, as well as classify them as pro- or counter-#MeToo. We then display the topic in each category with the highest weight α_i below.

protests, and supporting other participants in the #MeToo movement stays persistently prominent as the topics evolve over time. In addition, discussion around counter-#MeToo topics declined in prominence over time, and this discussion was subsumed into one topic by September 2019 (see Figures 1 and 2). In addition to the applications mentioned here, our method can be used to study a wide array of political science questions where large, unstructured data could provide powerful evidence toward extant theories. Providing empirical baselines could also help inform the generation of new theories.

2. Topic Models' Continued Usefulness in Political Science Research

Given the emergence of proprietary large language models (LLMs) and generative AI (ChatGPT and Claude), we demonstrate in this article that online topic model methods provide several advantages to political science researchers, such as being theoretically founded, open-source, and scalable. We emphasize that our methods are designed for data discovery, establishing new empirical facts, and helping to clarify new theoretical frameworks, especially for data that are unstructured. That said, we of course caution applied researchers from blindly applying this (or any) method. We encourage readers to view this method as an important first step to lead to deeper analysis, connecting to additional datasets, and informing new theories, especially where existing data lacked either granularity or real-time dynamics.

2.1. TLDA Provides Theoretical Foundations, Open Sourced Software, and Scalable Estimates

Our method has three key advantages over existing methods. First, our method has important theoretical properties. Neither the most popular LDA approaches based on Blei, Ng, and Jordan (2003) nor LLMs have yet been shown to have these theoretical foundations (Anandkumar *et al.* 2012, 2013, 2014). Our approach recovers—in feasible computational time—a provable identification guarantee for the topic–word probabilities and sample complexity bounds, as well as a form of statistical consistency in large samples (Anandkumar *et al.* 2012, 2013).[2] These foundations suggest that parameter recovery and

[2]Theorem 4.3 in Anandkumar *et al.* (2013) establishes that the spectral decomposition of the third-order tensor—and thus our method—accurately recovers topic–word probabilities, has no false positives, and accurately recovers the underlying topics in the data. Theorem 5.1 in Anandkumar *et al.* (2013) establishes finite sample complexity guarantees, a form of statistical consistency. In the case of TLDA, we have l_2 norm guarantees for each column of the topic–word probability matrix. Remark 11 in Anandkumar *et al.* (2013) notes that l_1 convergence, which is most familiar to political scientists, is not achievable in general.

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

large-sample accuracy are achievable when the text data follow a data generation process assumed by LDA, and that the topic–word probability matrix is full rank.

Second, our method is open-source, intentionally designed to be estimated on a wide array of workstations. Although LLMs show tremendous promise as a research method and clearly model human speech patterns more realistically than bag-of-word approaches like we propose, most of the popular LLMs are not open-source, cannot be trained locally without high-end computational resources (Linegar, Kocielnik, and Alvarez 2023), and demonstrate various social biases with significant implications for research using their model outputs, particularly on prompts concerning gender and race (Bartl, Nissim, and Gatt 2020; Kocielnik *et al.* 2023; Nozza, Bianchi, and Hovy 2021; Sheng *et al.* 2019; Zhao *et al.* 2018). By open-sourcing the training of the model, researchers can better test the sensitivity of their hyperparameter choices, run more robustness checks, and better assess the validity of their model outputs. While LLMs are extremely flexible and powerful and can be fine-tuned to a diverse array of tasks, this flexibility comes at the cost of an unconstrained, high-dimensional parameter space: the prompts needed to define the task that the model will perform. Small changes in these prompts often result in large and unpredictable differences in the model outputs, which make prompt-based methods very challenging to tune and reduce their reliability. The limiting principles of prompt engineering are an area of active research.

At the same time, in terms of interpretation and inference, proprietary LLM model weights, trained parameters, and specifics of training data are hidden from public view. For commercial LLMs, the model underpinnings and mechanics are areas of active research by those developing the LLMs. While it is possible for researchers to fix a particular version of an LLM to use in their research, it is not transparent to users how fixed any particular version of a commercial LLM might be. Unlike some proprietary models, our mathematical underpinnings are clearly and openly communicated and replicated, and the implementation itself is transparent and open source—previous versions are archived and available for reproduction purposes. Relatedly, due to their generative objective, commercial LLMs have been shown to "hallucinate," to literally shift their responses to prompts to completely different topics and subjects; by construction, our method will not hallucinate. Finally, by implementing a batched, streaming version of our method, researchers are freed from memory constraints that otherwise plague traditional unsupervised methods with large text corpora.

Third, both LLMs and supervised methods impose substantive financial resource hurdles that serve as barriers to their use by many researchers—hand labeling is expensive and proprietary LLM API calls can be extremely costly at scale. Furthermore, in applications where the entire population of documents contains valuable information, down-sampling may not be a viable solution to this cost. For example, 10,000 documents in a 200 million document sample (0.005%) might comprise a cogent and important topic, say congressional speeches opposing war, but due to sampling, such a critical topic may not be identified at all if only a few thousand documents are sampled. For both replication purposes and ready access to applied researchers, we hope our method allows political scientists to more readily answer questions of pressing concern, making more widespread use of large-scale text corpora numbering in the millions and billions of documents.

3. Building on Methodological Innovations in Political Science and Computer Science

In this section, we explain how we build from the foundation of two existing literatures. In the first case, we will leverage insights to build upon existing popular methods for topic models in political methodology. Second, we will contribute to a robust computer science literature on scalable LDA and Tensor LDA methods by proposing a GPU end-to-end pipeline for accurate and scalable open-source topic modeling.

However, in the case that words are not uniformly distributed within topics (and the word probabilities are concentrated in a small number of words within a topic), l_1 convergence can be achieved.

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

Table 1. Runtime of our TLDA method on GPU for 260 million and 1.04 billion documents using the COVID dataset.

	Time for each dataset (hours/minutes)	
Model	260 million docs	1.04 billion docs
Online TLDA	3h28.2	13h09

Note: None of the previous LDA methods scale to billions of documents.

3.1. Theoretical Guarantees Build on Existing Political Science Methods

Our contribution to the political methodology literature is to introduce topic model techniques from computer science that have statistical theoretical foundations. To cluster and analyze large text datasets, political science researchers make widespread use of unsupervised topic models, which do not always have parameter recovery and accuracy guarantees (Anandkumar *et al.* 2012, 2013). Among them, a popular model is LDA (Blei *et al.* 2003; Hoffman, Bach, and Blei 2010). This workhorse model can extract important information without requiring labeling or prior knowledge and has been used to analyze datasets across the social sciences, including studies on coordination among social movements, strategic communication of political elites, news dissemination, and the detection of toxic online behavior (Grimmer, Roberts, and Stewart 2022; King and Hopkins 2010; Lauderdale and Clark 2014). Also popular is the structural topic model (STM), which is closely related to LDA (Roberts *et al.* 2014; Roberts, Stewart, and Tingley 2016). These methods are popular due to the feasibility and easy implementation, but they often do not scale well to large datasets, so we turn to the computer science literature, where scale is key to unlocking new frontiers in research.

3.2. Feasible Improvements for Existing Scalable LDA

There have been numerous efforts to make LDA more scalable. Specifically, Yu *et al.* (2015) develop a method for faster sampling and distributing computation across multiple CPU cores. More recently, efficient GPU LDA implementations have been proposed: some have developed improved GPU workload partitioning (Wang *et al.* 2023; Xie *et al.* 2019), while Wang *et al.* (2023) developed a new branched sampling method. However, all of these implementations rely on traditional LDA methods based on Gibbs sampling, variational Bayes, or expectation maximization. The parallelization and scalability of these methods are inherently algorithmically challenging, as they are limited significantly by the sampling required to estimate the topics. Furthermore, unlike our method, these implementations are not available open-source, so they cannot be used easily by applied researchers.

Therefore, instead of relying on traditional LDA methods, we leverage tensor methods (Kolda and Bader 2009), which are embarrassingly parallel and have been proposed in order to scale to larger datasets (Papalexakis, Faloutsos, and Sidiropoulos 2016; Sidiropoulos *et al.* 2017). Using tensor methods, it is possible to learn latent variable models with accuracy guarantees under mild regularity conditions (Anandkumar *et al.* 2014; Janzamin *et al.* 2019). In particular, Tensor LDA relies on the computation of third-order moments, i.e., the three-way co-occurrence of words, and decomposes them to recover the topics. This approach has been shown to have similar performance as traditional LDA (Huang *et al.* 2015), which we also demonstrate empirically in the results of this article.

However, previous implementations of these tensor methods are limited by (1) the explicit construction of the second- or third-order moments and (2) a lack of hardware acceleration due to being developed either entirely for CPU or with a costly exchange between CPU and GPU. In particular, researchers have developed tensor LDA methods that face memory constraints due to the computation of high-dimensional low-order tensors (Anandkumar *et al.* 2012, 2013, 2014). Further, these methods only run on CPU and do not benefit from hardware acceleration on GPU. But Huang *et al.* (2015) developed a stochastic tensor gradient descent (STGD) approach to estimate the third-order decomposition, allowing for further scaling of the method. However, this method has CPU–GPU exchange and relies on the explicit construction of the second-order moment, so it cannot be run fully online.

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

Similarly, Swierczewski *et al.* (2019) proposed a method for learning the third-order decomposition using alternating least squares but is limited by a CPU-based implementation.

By contrast, we derive an efficient centered, online version of the TLDA that scales linearly to any dataset size (with constant memory). We provide evidence supporting this claim in Table 1. We provide an end-to-end GPU-accelerated implementation that will be open-sourced along with this article to enable its application to any dataset by other researchers.

4. How to Achieve Scalable Tensor LDA

We provide a summary overview of our method in Figure 3. As documents are provided as input to our TLDA pipeline, they are first pre-processed and converted into bag-of-words vectors. The key to TLDA methods is to perform no more dimension reduction than needed to ensure the method has theoretical accuracy guarantees, but to perform enough dimension reduction to achieve scale. Anandkumar *et al.* (2013) demonstrated that two dimension reductions will produce accurate results. Similarly, our method takes the document–topic frequency matrix and takes the average word frequency across documents. We call this average $\boldsymbol{M}_1$, the first moment. Then, we can demean the data and by doing so, we automatically update the model as we stream in new documents. Demeaning the data is a very powerful tool for reducing computational complexity because it cancels out very ugly off-diagonal terms for the higher-order moments that we calculate in our model. By canceling out these terms, we can now stream data into the method and automatically update the results. Computer scientists call this an online update, terminology we adopt here and throughout the text (online centering and update of $\boldsymbol{M_1}$ in Figure 3). We then perform our first dimension reduction on the word co-occurrence matrix $\boldsymbol{M_2}$ to find singular values, so we can use them to transform ("whiten") the data. Whitening is a linear transformation that reduces the dimensionality of the data from the number of words to the number of topics.[3] The word-occurrence matrix simply records how many times words occur together in the corpus. Then, rather than directly compute the singular values for the word-occurrence matrix to get the whitening values, we can implicitly calculate them by performing PCA on the demeaned (centered) data, $\tilde{\boldsymbol{x}}$ (online whitening and update of $\boldsymbol{M_2}$ in Figure 3). Now, we take the whitening values and use them to transform the demeaned (centered) data. This "whitening" step reduces the size of the data from $V \times V \times V$ (number of words by number of words) to $K \times K \times K$, the number of topics. This vastly reduces the size of the problem because in every real-world application, the number of topics will be vastly smaller than the number of words. Having performed this transformation, our data have been reduced from size $V \times M$ to size $K \times M$. We now compute the "whitened" analog of the word tri-occurrence matrix, $\boldsymbol{M_3}$, which is $K \times K \times K$. We then find the eigenvalues of this $K \times K \times K$ object, which after some algebraic transformations, we show are equivalent to the topic model outputs from LDA (the unwhitened, uncentered learned factor in Figure 3).

Here, we propose several improvements over previous work to enable scaling the TLDA to billions of documents. First, we reduce the computation complexity of the second- and third-order cross moments; we derive them on centered data. Second, we incrementally estimate PCA on the centered documents. We take these principal components and use them to implicitly form the decomposition of the second-order moment.[4] We derive a simplified, batched gradient update, leading to efficient recovery of the decomposition of the third-order moment. We jointly learn all moments online by updating the mean, PCA, and third-order decomposition in one pass, instead of relying on a precomputed dimensionality reduction of the second-order moment.

[3] Whitening is a linear transformation that produces a new data matrix where each column is a de-correlated topic and the variance is standardized to 1.

[4] We leverage an algebraic relationship between principal components of the first moment and the SVD on the second moment to extract the whitening matrix without needing to fully calculate the SVD of the second-order moment, which is computationally taxing. This implicit calculation allows us to parsimoniously decompose the data. This is helpful because the uncentered data are no longer sparse, increasing the memory footprint of the dataset.

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

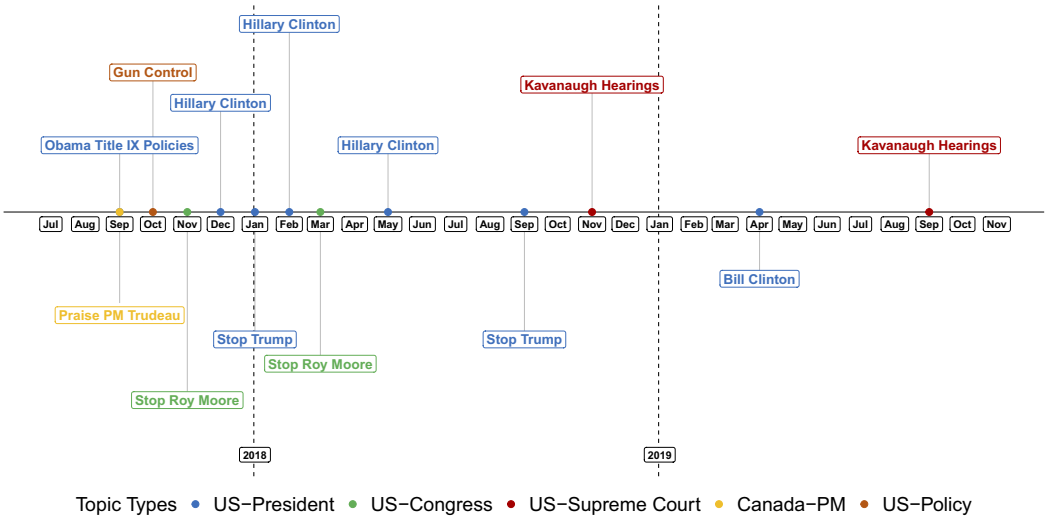


Figure 2. Evolution of most prominent political topics in the #MeToo discussion.
Note: In each iteration of the dynamic analysis detailed in Section 7.2, we inspect the topics, manually label them, and classify them as political or not political. We display the political topic with the highest weight α_i below.

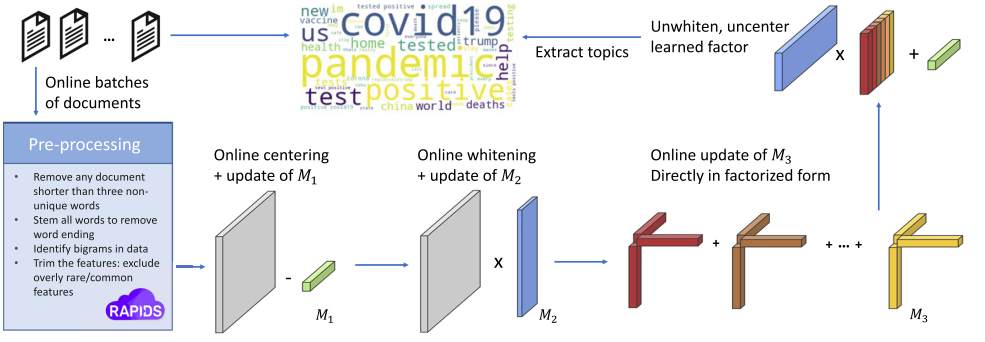


Figure 3. Overview of our approach.
Note: As batches of documents arrive, incrementally, they are first pre-processed (they are stemmed, tokenized, and the vocabulary is standardized). We then create a dataset of the counts for each word in each document. We then find the average number of times each word appears in each document (the average word occurrence, which is the first moment M_1) and subtract the value of M_1 from our existing word-frequency matrix. The resulting document term matrix is our centered dataset, X (Section 4.1). We then perform a singular value decomposition on the centered data, X, to recover whitening weights without ever needing to calculate M_2, directly. This saves computational overhead while being mathematically equivalent. We then use these whitening weights to transform the centered data, X, which can be done incrementally (Section 4.3). Finally, we construct the whitened equivalent of the third-order moment, M_3, which is updated, directly in this factorized form (Section 4.4). This learned factorization can be directly unwhitened and uncentered to recover the classic solution to TLDA (Section 1) and recover the topics and their associated word probabilities (Section 4.6).

Crucially, we prove that the original topics computed in prior spectral LDA work can be recovered from the topics produced by our method with a simple post-processing step. As a result, our method enjoys the theoretical benefits of spectral LDA while significantly reducing its computational complexity. After finding the topic distribution over words, we employ standard variational inference (VI) to recover document-level parameters. We propose an efficient implementation on GPU within the TensorLy framework, which can easily scale to very large datasets. We preview the substantive results of this method, displaying the most prominent topics over time as identified by the learned weights, which is possible thanks to our tensor-based approach (Figure 3).

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

4.1. Computing the Centered Cumulants

In this section, we provide a technical overview of the method. For a summary of our notation and the mathematical background for the method, please refer to the Supplementary Material (Table A1). We begin by introducing the model, the data generation process, and the estimation routine.

The data are a corpus of documents which we note in the following way. Let the data be a document term matrix $\boldsymbol{F}$ with rows $\boldsymbol{f}_t := (f_{t,1}, f_{t,2}, \ldots, f_{t,V}) \in \mathbb{R}^V$ denoting the vector of word counts for the t-th document, where V is the number of words in the vocabulary, and let N be the number of documents. Finally, we will let K denote the total number of topics and h be the topic labels. Then, the first-order moment is

$$\boldsymbol{M}_1 := \frac{1}{N}\sum_{i=1}^{N}\boldsymbol{f}_i. \tag{1}$$

Our first innovation is to center the data by forming $\tilde{\boldsymbol{x}}_i := \boldsymbol{f}_i - \boldsymbol{M}_1$. Given this set-up, we can simplify the usual moments for spectral LDA (see Anandkumar *et al.* 2012, 2013, 2014, as well as Section 1.2 of the Supplementary Material). This is because the diagonalization terms in $\boldsymbol{M}_2$ and $\boldsymbol{M}_3$ become $\mathbf{0}$ in expectation, as the first moment is now $\mathbf{0}$ for the centered data matrix $\tilde{\boldsymbol{X}}$. This vastly reduces the number of off-diagonal calculations required to estimate the higher-order moments.

By removing the corresponding terms that are now $\mathbf{0}$ in expectation from the expression of the moments in Anandkumar *et al.* (2013), we now have the following simplified empirical moments for the centered data $\tilde{\mathbf{X}}$, where α_0 is the topic mixing parameter

$$\tilde{\mathbf{M}}_2 := \frac{(\alpha_0+1)}{N}\sum_{t=i}^{N}\tilde{\boldsymbol{x}}_i \otimes \tilde{\boldsymbol{x}}_i \tag{2}$$

$$\tilde{\mathbf{M}}_3 := \frac{(\alpha_0+1)(\alpha_0+2)}{2N}\sum_{i=1}^{N}\tilde{\boldsymbol{x}}_i \otimes \tilde{\boldsymbol{x}}_i \otimes \tilde{\boldsymbol{x}}_i, \tag{3}$$

where $\otimes$ is the Kronecker product.

Following Anandkumar *et al.* (2013), we know that the moments for the centered data can be factorized as

$$\mathbb{E}\left[\tilde{\boldsymbol{M}}_1\right] := \mathbf{0} \tag{4}$$

$$\mathbb{E}\left[\tilde{\boldsymbol{M}}_2\right] := \sum_{i=1}^{K}\frac{\alpha_i}{\alpha_0}\boldsymbol{\nu}_i \otimes \boldsymbol{\nu}_i \tag{5}$$

$$\mathbb{E}\left[\tilde{\boldsymbol{M}}_3\right] := \sum_{i=1}^{K}\frac{\alpha_i}{\alpha_0}\boldsymbol{\nu}_i \otimes \boldsymbol{\nu}_i \otimes \boldsymbol{\nu}_i, \tag{6}$$

where $\tilde{\boldsymbol{M}}_1$, $\tilde{\boldsymbol{M}}_2$, and $\tilde{\mathbf{M}}_3$ are the first, second, and third moments of the centered data, and $\boldsymbol{\nu} = [\boldsymbol{\nu}_1, \ldots, \boldsymbol{\nu}_K]$, the learned decomposition of $\tilde{\mathbf{M}}_3$. Note that we use the centered analog from Anandkumar *et al.* (2013), which showed that the singular value decomposition of the third-order moment tensor yields estimates for the LDA model parameters.

Finally, to recover the actual topic–word probability matrix as derived in Anandkumar *et al.* (2013) using the topics computed from the centered data, we prove Theorem 1, demonstrating that the ground-truth factors can be recovered by de-centering the factors from the centered ones.

Theorem 1. *Given the factors $\boldsymbol{\nu}_i$ learned from the centered data, we show that*

$$\mathbb{E}[\mathbf{M}_3] = \sum_{i=1}^{K}\frac{\alpha_i}{\alpha_0}(\boldsymbol{\nu}_i + \boldsymbol{M}_1) \otimes (\boldsymbol{\nu}_i + \boldsymbol{M}_1) \otimes (\boldsymbol{\nu}_i + \boldsymbol{M}_1).$$

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

That is, the true third-order cumulant can be recovered directly by re-centering the factors of the decomposition of the centered cumulant, indicating that the vectors $\boldsymbol{\nu}_i + \boldsymbol{M}_1$ are equivalent to the ground-truth factors from Anandkumar et al. (2013).

For the proof, see Section 1.3 of the Supplementary Material.

Following from Theorem 1, we have

$$\begin{aligned}\mathbb{E}\left[\mathbf{M}_3\right] &= \sum_{i=1}^{K} \frac{\alpha_i}{\alpha_0} (\boldsymbol{\nu}_i + \boldsymbol{M}_1) \otimes (\boldsymbol{\nu}_i + \boldsymbol{M}_1) \otimes (\boldsymbol{\nu}_i + \boldsymbol{M}_1) \\ &= \sum_{i=1}^{K} \frac{\alpha_i}{\alpha_0} \boldsymbol{\mu}_i \otimes \boldsymbol{\mu}_i \otimes \boldsymbol{\mu}_i,\end{aligned}$$

where $\boldsymbol{\mu} = [\mu_1, \ldots, \mu_K]$ and $\boldsymbol{\mu}_i = Pr(f_j | h = i)$, where $\boldsymbol{h}$ are the set topic labels, i is the i'th topic, and j is the j'th word in the vocabulary. In other words, $\boldsymbol{\mu}$ is the topic word matrix of the uncentered data $\boldsymbol{F}$.

4.2. Batched Implementation Enables Arbitrary Scaling

In the sections below, we present a batched implementation for TLDA. Each moment is individually calculated incrementally as we feed in data to the method, and then we estimate the topic–word probabilities using stochastic gradient descent.

In the section that follows the online decomposition for the second and third moments, we present a fully online implementation, which we recommend for very large datasets on the scale of billions of documents. For such datasets, the individual contribution of one data point is extremely small, so we average over many documents with minimal loss to accuracy given the enormous gains to scale.[5] In practice, this means we can iterate through the documents just once to still achieve our convergence criteria and to achieve accurate inference. As a byproduct of this implementation, we can then update the model by streaming new data points into the training API, giving a means to offer a fully online version of the model as part of this library.

4.3. Online Decomposition of the Second Moment

As a function of centering the data, our method streamlines the pipeline that Anandkumar *et al.* (2012) proposes for calculating the second moment and the whitening matrix. Specifically, instead of constructing $\tilde{\boldsymbol{M}}_2$, which is very memory-intensive for large data, we implicitly form $\tilde{\boldsymbol{M}}_2$ by computing a singular value decomposition of the centered data matrix.

Using the singular values and singular vectors from the centered data, we construct a whitening matrix $\boldsymbol{W}$ such that

$$\boldsymbol{W}^\top \tilde{\boldsymbol{M}}_2 \boldsymbol{W} = I. \tag{7}$$

We let D be the whitening dimension size. We note that from Anandkumar *et al.* (2012, 2013, 2014), letting $D = K$ is sufficient to compute the third-order decomposition, although a slightly larger D can be chosen to improve the dimensionality reduction. Then, from the centered data, we have

$$\boldsymbol{W} = \frac{\sqrt{\alpha_0 + 1}}{N} \boldsymbol{U} \boldsymbol{\Sigma}^{-\frac{1}{2}},$$

where $\boldsymbol{U}$ and $\boldsymbol{\Sigma}$ (the variance matrix of the centered data) are the top D singular vectors and singular values of the centered data, obtained through computing the PCA of $\tilde{\boldsymbol{X}}$, which is equivalent to its SVD since the data are centered.

[5] For datasets on a smaller scale, individual documents can have much larger implications for topical inference, so the batched version is preferred, as empirical tests show that multiple loops through the data are necessary to meet the convergence criteria for the third-order moment and for accurate inference.

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

Then, the $\tilde{\boldsymbol{M}}_3$ tensor is implicitly formed using the whitened counts of the centered data. Whitening renders the tensor symmetric and orthogonal (in expectation). Most importantly, it reduces the dimensionality of the third moment from size N^3 to $D^3 \approx K^3$, where K is the number of topics. Given the nature of speech in social environments, the number of topics will almost always be at least an order of magnitude smaller than the number of words.

To estimate the implicit third moment, the method calculates the whitened counts $\boldsymbol{x} = \boldsymbol{W}\tilde{\boldsymbol{X}}$. We will use these whitened counts to construct the implicit third-order tensor. Using that implicit tensor, the method utilizes stochastic gradient descent to find the spectral decomposition of the third-order moments.

4.4. Online Learning of the Third Moment

Here, we formulate the TLDA framework in a vectorized form, solving for a *batch* of data, resulting in a much more efficient implementation. Let $\boldsymbol{\Phi} = [\Phi_1|\Phi_2|...|\Phi_K]$ be the eigenvectors of the third-order moment for the whitened, centered data. We note that each eigenvector Φ_i is of length D and denote the full sample size as N.

Now, note that the decomposition of the third-order moment for the whitened, centered data $\boldsymbol{X}$ (of size $D \times D \times D \approx K \times K \times K$) is

$$\boldsymbol{T} = \sum_{i \in K} \boldsymbol{\Phi}_i \otimes \boldsymbol{\Phi}_i \otimes \boldsymbol{\Phi}_i.$$

With the whitened tensor in hand, the method follows Huang *et al.* (2015) in implementing a batched STGD algorithm for tensor CP decomposition.

Specifically, we consider a mini-batch of n_B centered and whitened samples $\boldsymbol{x}_1, \ldots \boldsymbol{x}_{n_B}$, which we collect in a matrix $\boldsymbol{X} \in \mathbb{R}^{n_B \times D}$. We want to learn a tensor factorization of the third-order whitened and centered cumulant:

$$\begin{aligned} \tilde{\boldsymbol{M}}_3 &= \frac{(\alpha_0+1)(\alpha_0+2)}{2N} \sum_{n=1}^{N} \boldsymbol{x}_n \otimes \boldsymbol{x}_n \otimes \boldsymbol{x}_n \\ &= \frac{(\alpha_0+1)(\alpha_0+2)}{2N} \sum_{n=1}^{N} \boldsymbol{x}_n \otimes^3 . \end{aligned} \tag{8}$$

We are trying to learn a rank-K CP factorization with factors $\boldsymbol{\Phi}$ of $\tilde{\boldsymbol{M}}_3$ such that $\tilde{\boldsymbol{M}}_3 = \mathbf{T} = \sum_{i=1}^{K} \boldsymbol{\Phi}_i \otimes \boldsymbol{\Phi}_i \otimes \boldsymbol{\Phi}_i$. In other words, we solve the following optimization problem:

$$\underset{\boldsymbol{\Phi};\, ||\boldsymbol{\Phi}_i||_F^2 = 1}{\arg\min} \underbrace{||\tilde{\boldsymbol{M}}_3 - \mathbf{T}||_F^2}_{\text{reconstruction loss}} + \underbrace{\frac{(1+\theta)}{2} ||\mathbf{T}||_F^2}_{\text{orthogonality loss}}. \tag{9}$$

In plain words, we minimize the reconstruction loss while inducing orthogonality on the decomposition factors. This can be seen by noting that the factors (and therefore the rank-1 components) are normalized, meaning the Frobenius norm of the second term simplifies to only the inner product between the components.

The problem in equation (9) thus simplifies to

$$\begin{aligned} \underset{\boldsymbol{\Phi};\, ||\boldsymbol{\Phi}_i||_F^2 = 1}{\arg\min} & \frac{(1+\theta)}{2} ||\sum_{i=1}^{K} \boldsymbol{\Phi}_i \otimes^3 ||_F^2 \\ & - \langle \sum_{i=1}^{K} \boldsymbol{\Phi}_i \otimes^3, \frac{(\alpha_0+1)(\alpha_0+2)}{2N} \sum_{n=1}^{N} \boldsymbol{x}_n \otimes^3 \rangle. \end{aligned} \tag{10}$$

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

This can be equivalently written in matrix form using the Khatri–Rao product:

$$\underset{\boldsymbol{\Phi};\|\boldsymbol{\Phi}_i\|_F^2=1}{\arg\min} \frac{(1+\theta)}{2}\|\boldsymbol{\Phi}(\boldsymbol{\Phi}\odot\boldsymbol{\Phi})^\top\|_F^2 - \frac{(\alpha_0+1)(\alpha_0+2)}{2n}\langle\boldsymbol{\Phi}(\boldsymbol{\Phi}\odot\boldsymbol{\Phi})^\top, \boldsymbol{X}^\top(\boldsymbol{X}\odot\boldsymbol{X})\rangle. \tag{11}$$

By taking the derivative with respect to the decomposition factor $\boldsymbol{\Phi}$, we get

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{\Phi}} = 3(1+\theta)\boldsymbol{\Phi}(\boldsymbol{\Phi}^\top\boldsymbol{\Phi} * \boldsymbol{\Phi}^\top\boldsymbol{\Phi}) - \frac{3(\alpha_0+1)(\alpha_0+2)}{2n}\boldsymbol{X}^T(\boldsymbol{X}\boldsymbol{\Phi} * \boldsymbol{X}\boldsymbol{\Phi}). \tag{12}$$

We then update the factor via batched stochastic gradient update:

$$\boldsymbol{\Phi}_{t+1} = \boldsymbol{\Phi}_t - \beta\frac{\partial\mathcal{L}}{\partial\boldsymbol{\Phi}}, \tag{13}$$

with β as the learning rate.

4.5. Fully Online Implementation

In a fully batched implementation above, the method relies on computing each higher-order moment sequentially, even though each moment is individually learned online. By contrast, in the fully online version of our TLDA method presented here, we learn both moments by jointly learning both the second- and third-order moments online.[6] We first find initial values for the factors by running the batched TLDA to convergence on a small portion of the data. Then, when given a new batch of data, we first update $\boldsymbol{M}_1$, then update the incremental PCA (and use the new version of the PCA to whiten the data), and finally perform a gradient update of the third-order moment using the new batch of whitened data. As a result, we can update the third-order moment decomposition only using the new batch without looping through any prior data. This is in contrast to the batched version of our method, where we loop through the entire dataset three times to compute the first-order moment, second-order moment decomposition, and third-order moment decomposition, respectively.

Although we only perform one gradient descent step per batch of data in this version of the method, we expect that for large datasets, there are sufficiently many documents so that this does not significantly impact the quality of the factors produced.[7] We use online LDA to obtain topic coherence values that are similar to or better than existing methods.

4.6. Recovering the Topic Model Parameters

Once we have learned the factorized form Φ of the third-order moment, we describe how we recover the uncentered, unwhitened moment to recover the topics. First, we obtain $\boldsymbol{\nu}$, the estimate of the decomposition of $\tilde{\boldsymbol{M}}_3$, by unwhitening the components Φ of the decomposition:

$$\boldsymbol{\nu} = \boldsymbol{W}^{T^\dagger}\boldsymbol{\Phi},$$

where † denotes the pseudo-inverse. Using this, we can find

$$\alpha_i = \gamma^2\boldsymbol{\nu}_i^{-2}.$$

Here, γ is a scaling factor such that $\sum_{i=1}^k \alpha_i = 1$.

As derived in Theorem 1, we can then re-center $\boldsymbol{\nu}$ to compute the estimate for the uncentered topic–word probabilities

$$\boldsymbol{\mu}_i = \boldsymbol{\nu}_i + \boldsymbol{M}_1.$$

[6] The first moment is just the average word frequency and trivial to compute at scale.

[7] We confirm this claim empirically in Section 5.

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

5. The TensorLy-LDA Package

Along with this article, we release a new Python package that provides an efficient, end-to-end GPU-accelerated implementation of our proposed online Tensor LDA.[8] We used that implementation for all experiments in this article. It consists of two main steps: an efficient pre-processing module that uses RAPIDS and a module that builds on top of TensorLy to learn the higher-order cumulants.

Our entire Tensor LDA method is end-to-end GPU accelerated and implemented on the Nvidia RAPIDS Data Science Framework, a GPU-based architecture for data analysis in python (RAPIDS Development Team 2018), as well as the TensorLy library, a high-level API for tensor methods in Python (Kossaifi *et al.* 2019). First, the data are pre-processed, on GPU, using RAPIDS. After the data have been pre-preprocessed, all tensor operations are performed using the TensorLy library, which is used to learn the third-order cumulant in factorized form directly. RAPIDS is used for learning the second-order cumulant through incremental PCA. The result is an end-to-end GPU implementation of a large-scale topic model with no CPU–GPU exchange. We empirically establish our implementation in the next section through thorough experiments and demonstrate large speedups over previous works.

The library provides all the tools to run our method on an actual dataset. To facilitate adoption by practitioners, it comes with a thorough online documentation and interactive examples. Both the examples and an extensive suite of unit tests are run dynamically after any change in the codebase through a continuous integration suite to ensure correctness.

5.1. Input and Output

The method takes a data matrix where each entry is the centered word frequencies for each of V words for N documents. Each column represents a word in the dataset and each row is a document, for a matrix of size $N \times V$. The method produces two key outputs: first, the topic–word matrix and second, the learned weights, α_i. These outputs can be fed as inputs into a standard VI method that calculates the topic–document matrix, as well. We have included a standard VI method in our API so users can calculate the document–topic matrix for their applications.

Hyperparameter Tuning

There are several hyperparameters standard to LDA and STGD methods that are under researcher discretion. Users are encouraged to check that the default parameters are appropriate for their application.

1. Number of topics k: number of learned clusters. Should be optimized by a researcher.
2. Topical mixing, α_0: The level of mixing believed to be in the documents. Closer to 0 is no mixing and closer to ∞ means fully mixed documents.
3. Learning rate β. How much to allow new batches of data to contribute to the factor update. Needs to be tuned for stable convergence (if convergence is too slow, increase it. If topics appear noisy or nonsensical, decrease it).
4. Orthogonality penalty θ: How much separation you expect between topics. If topical mixtures appear too similar, increase this parameter. If topics are incoherent or convergence is unstable, decrease it.

Recommendations for Data Pre-Processing

By pre-processing the data on the Rapids GPU framework, we alleviate a crucial bottleneck in the practicability of LDA on large datasets. Although pre-processing has been shown to be critical to producing valid results, especially in social science contexts (Grimmer and Stewart 2013), existing frameworks for topic models often entail expensive CPU–GPU exchange. Having overcome this bottleneck, we follow best practices suggested by Grimmer and Stewart (2013) and summarized in King

[8] The package installation instructions can be found on its website, here: https://tensorly.org/tlda/dev/install.html.

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

and Hopkins (2010); we optimize feature selection by stemming and tokenizing the data. The political science literature has found that for non-noisy inference on text data, we want neither too few common features such that there is no variation amongst documents nor too many uncommon features such that there are no distinguishable clusters.

We follow this process to arrive at our final set of features:

- Remove any document shorter than three non-unique words.
- Stem all words to remove word endings using a Porter Stemmer.
- Identify bigrams in the data.
- Trim the features: exclude any features appearing in fewer than the lower bound that scales with the number of documents of the document and more than the upper bound that scales with the number of documents.

The political science literature has intensively explored the sensitivity of critical substantive findings to pre-processing. King and Hopkins (2010) find that the consensus in the social science literature is that brute force unigram-based methods, with rigorous empirical validation, will typically account for the majority of the available explanatory power in the data. So long as pre-processing captures all relevant features, our inferences derived from NLP can be used to analyze social phenomena. However, King and Hopkins (2010) note that the tuning of pre-processing choices generally depends on the nature of the application. In all of the applications presented in this article, our unit of observation is a tweet, an inherently short document limited to 270 characters. Finally, following standard practice for topic models, we stem words to their base root so that the core meaning of these words is captured by only one token. In previous applications, this has been shown to both improve computational tractability and clarify the substantive analysis of text data.

5.2. Package Availability

The package is fully open-source as part of the TensorLy (Kossaifi *et al.* 2019) project, under the BSD-3 license, which makes it suitable for any use, academic or industrial. It is well tested and has extensive documentation.

6. Simulations

6.1. Parameter Recovery and Comparison to Previous TLDA Methods

In this section, we demonstrate that our method results in gains in accuracy, topic correlation, and speed in comparison to existing TLDA methods in a simulated setting. We use the traditional LDA Data Generation Process for generating the simulated data (see Section 1.4 of the Supplementary Material and Blei *et al.* 2003). We present a comparison to two key existing versions of the TLDA method: (1) the spectral decomposition algorithm in Anandkumar *et al.* (2012, 2013, 2014) and (2) the SGD-based TLDA derived in Huang *et al.* (2015), in which the third-order moment is computed online. To do so, we show comparisons of all three TLDA methods on synthetic data. Due to the small scale of the synthetic experiment (20,000 documents), we run the batched version of our method.

As a result, through this process, we obtain ground-truth factors that adhere to the assumptions of LDA. By running TLDA on the synthetic document vectors, we can then compare each factor in the learned topic–word matrix to the corresponding factor in the generated ground-truth topic–word matrix by computing their correlation.

However, the topics in the learned topic–word matrix can be in any order, so we limit the number of topics to $K = 2$ and use the permutation that maximizes the average correlation to the ground truth. We use parameters $\alpha_0 = 0.01$ and whitening size $D = 2$ for all TLDA methods, as well as learning rate 1×10^{-4} for the SGD-based TLDA and our method. Table 2 shows the correlation between the learned and ground-truth factors in corpora with 20,000 documents. The results are averaged over ten random seeds for each combination of parameters. This table illustrates that under a variety of vocabulary sizes, our method is more accurate than existing tensor methods, as evidenced by the higher mean and lower standard deviation of correlations among all runs.

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

Table 2. Comparison of topic recovery on synthetic data for various TLDA methods.

	Average correlation			Standard deviation of correlation		
Vocabulary size	PARAFAC TLDA[1]	SGD TLDA[2]	Batched TLDA[3]	PARAFAC TLDA[1]	SGD TLDA[2]	Batched TLDA[3]
500	0.886	0.893	**0.943**	0.304	0.306	**0.115**
1,000	0.906	0.787	**0.930**	0.183	0.404	**0.090**
1,500	0.872	0.840	**0.873**	0.265	0.363	**0.101**

Note: (1) Anandkumar *et al.* (2014); (2) Huang *et al.* (2015); and (3) Our method. The method achieving the highest average correlation and lowest standard deviation among the correlations is bolded at each vocabulary size.

Table 3. Comparison of CPU runtime on synthetic data for various TLDA methods.

	Average time (s)		
Vocabulary size	PARAFAC TLDA[1]	SGD TLDA[2]	Batched TLDA[3]
500	20.00	53.87	3.00
1,000	33.40	71.75	2.98
1,500	62.18	103.39	3.55

Note: (1) Anandkumar *et al.* (2014); (2) Huang *et al.* (2015); and (3) Our method.

In Table 3, we compare the average runtime of the three TLDA methods for the synthetic data experiments in Table 2. In Table A4 in the Supplementary Material, we also compare against other scalable LDA methods—we note our TLDA method compares favorably and that none of the methods against which we compare have theoretical accuracy guarantees. We run this analysis on the CPU due to the relatively small scale of the experiment. Our version of the TLDA method results in a runtime that is between 6 and 20 times faster than the existing TLDA methods. Furthermore, as shown in Table 3, as the vocabulary size increases, the runtime of our method increases its relative speed advantage over the others. This demonstrates the value of the simplifications we introduce in the method section; we significantly outperform the existing TLDA methods in terms of runtime while also making non-trivial gains in accuracy.

7. Applications

In this section, we demonstrate the scalability of our method by applying it to two large-scale Twitter datasets—concerning the #MeToo movement and the 2020 U.S. Presidential elections. These applications present analyses of important datasets that political scientists might wish to use to study collective action, political behavior, gender politics, election misinformation, and many other theoretically- and substantively-important topics. But analyses like these would have been infeasible or perhaps impossible due to the large size of the data (as we demonstrate below) without methods like TLDA.

The #MeToo dataset comprises 7.97 million tweets related to the political and social discussion surrounding #MeToo. We conduct thorough ablation studies using the #MeToo dataset and empirically demonstrate that the runtime of our method scales linearly as the number of documents increases and is nearly constant as the number of topics increases. We compare the runtime and topic coherence to a popular off-the-shelf model to show that our online method is 15×–140× faster than previous methods while achieving similar or better coherence.[9] Additionally, we show the practical utility of our method for applied researchers by using it to dynamically analyze the evolution of the #MeToo dataset over time. We show qualitative evidence of topical evolution in the discussion around the social movement and political coordination on social media.

[9]The machines used for all timing experiments are reported in Table A5 in the Supplementary Material.

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

The second application uses a dataset of approximately 29 million tweets that were collected during the 2020 presidential election, regarding the conduct of the election. We show below that our method can process and analyze these data quickly and efficiently, generating interesting topic modeling results that could shed light on important political science questions.

Finally, in the Supplementary Material, we analyze a third dataset that contains over 260 million tweets related to COVID-19, collected in real-time using keywords from the Twitter streaming API. We show that our method produces coherent estimates in under 3.5 hours on this dataset, and in under 13.5 hours on a simulated 1.04 billion document dataset created using these COVID-19 data. Thus, we demonstrate that our method is effective for unsupervised analysis of large-scale data on the order of billions of documents.

7.1. The #MeToo Movement: Scaling, Ablation, and Substantive Studies

7.1.1. Studying Mass Movements and Collective Action with Large-Scale Datasets

The #MeToo movement is a prolific women's rights movement that gained traction extremely quickly on Twitter in October 2017, with over 7.9 million tweets containing the #MeToo hashtag from October 2017 to October 2018 alone (Brown 2018). This movement is an important example of what Clark-Parsons has termed "Network Feminism," where social media platforms have become a crucial organizational tool for mobilization of social and political movements (Clark-Parsons 2022).

Going back to the early theoretical work of Mancur Olson, studying social movements and protest politics as a lens for collective action has been an important literature in political science (Olson Jr. 1971). In particular, researchers have long tried to understand the political origins of protest politics and mass movements, because as Olson noted, participation can be costly and the results of participation can be difficult for individuals to assess.

Studying how mass movements and protest politics arise, how they are organized, and how they are sustained in the long run, is also complicated by a lack of available data. Movements and protests arise quickly, and authorities often act to stop and prevent protesting and organizing, which means in many cases that political scientists cannot often collect data about protests and movements. Surveys of protest participants can be done after the fact, but they can be difficult to find, difficult to persuade to participate in a survey, and their survey responses may be inaccurate with the passage of time. Thus, much of the literature has resorted to case studies of historical examples (Chong 1991).

In recent years, the use of social media by protest and movement participants has sparked new research about collective action, specifically regarding protests and mass movements. By collecting data from participants in the protests, while they are protesting or acting collectively, has proven to be an important way to generate datasets to test existing theories, for example, about the Arab Spring or Black Lives Matter movements (Kann *et al.* 2023; Steinert-Threlkeld 2017).

In a similar way, the tweets related to #MeToo thus provide rich data for investigating the evolution of a modern social movement initiated by online discussions on social media. What topics engaged participants in the #MeToo movement? Did the topics of conversation change over the course of the movement? Can the language of social media conversations help us determine the motivations of participants in the movement, where they motivated by self-interest or collective concerns? These data can be crucial for understanding this important social and political movement, and for testing theories about how movements like these arise and are sustained.

We analyze the topics present in a corpus of #MeToo tweets collected from January 2017 to September 2019, which contains 7.97 million tweets after pre-processing. Figure 4 shows the initial proliferation of tweets in January 2017 related to the Women's March and Movement, as well as the viral growth of the #MeToo movement in September and October 2017.

7.1.2. Pre-Processing

We follow the standard pre-processing framework outlined earlier in the article, which we note takes only 178 seconds on GPU using RAPIDS. The #MeToo Twitter data are high-frequency, and there

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

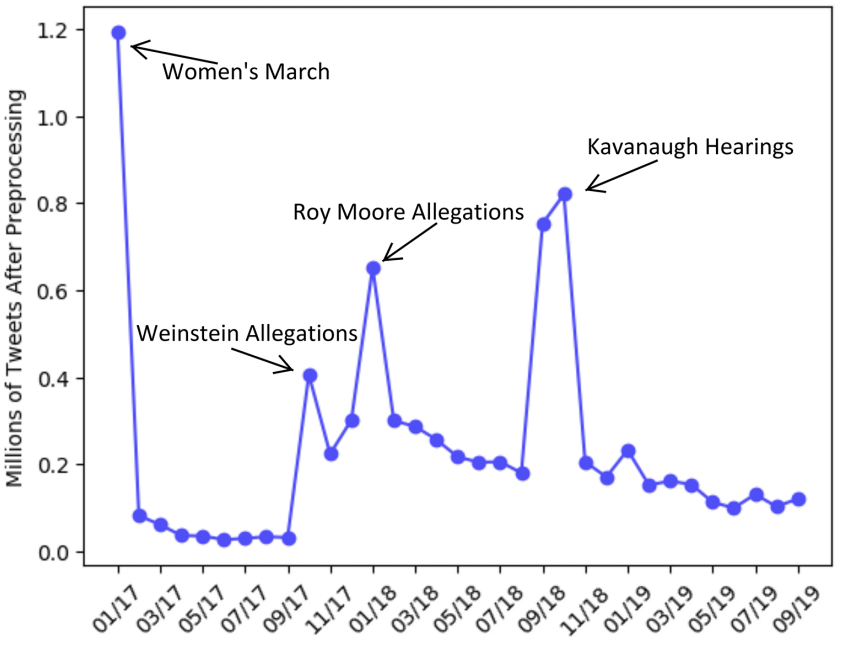


Figure 4. Tweets per month in the #MeToo data, in millions.

are orders of magnitude many more observations than in existing datasets analyzed in applications in the social science literature. Thus, we set a very low lower bound for pruning words—words need only appear in 0.002% of documents to be included in the #MeToo data. Higher thresholds would dramatically reduce the number of features such that there was no variation in the document structure. Any lower and many features which only appear in a handful of documents would be essentially noise. These are proper nouns, such as usernames, typos, nonsensical words, or words that are not common enough to help define meaningful clusters. At the same time, we exclude words that appear in more than 50% of the documents. This only amounts to removing 50 words, many of which are so common as not to be useful in delineating a topic due to lack of variation (as they appear in every document). We arrive at 1,837 words, more than enough to pin down meaningful topics. Changing this cutoff to 70%, 80%, or 90% does not significantly change the number of words in the vocabulary. Still, we stick to the more restrictive cutoff because many common words would otherwise dominate every topic, hindering the interpretability of the model. We also ensure that words occur in at least 0.002% of documents, so idiosyncratic words that explain slight variation in the data are excluded. These words might appear in only one or two documents, which is far too infrequently to pin down substantive topics. Finally, following standard practice, we stem words by cutting off verb and noun endings so that base words will carry the same semantic meaning.

7.1.3. #MeToo Scaling and Convergence Speed Comparison

In this set of results, we first run the batched and online versions of our TLDA method on the entire #MeToo Twitter dataset on one GPU core to analyze how quickly both versions of our approach converge with varying numbers of topics. We then compare the scaling of our TLDA method to that of Gensim by computing convergence time on subsets of the #MeToo dataset containing 1M, 2M, 5M, and 7.97M documents.

Full #MeToo Timing Comparison To choose the optimal parameters for our method, we run an extensive grid search over the number of topics K, whitening sizes D, topic mixing parameters α_0, and learning

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

rates β. For each number of topics, we determine the optimal parameters by finding the parameter combination with the highest mean coherence score across all metrics. For $K = 10$, we report the 20 top words for the batched model with the best parameters in Table A2 in the Supplementary Material. We include the parameters used for each number of topics in Table A6 in the Supplementary Material.

Using these optimal parameters, we run the batched and online versions of our model to convergence and provide a benchmark comparison of speed and coherence on GPU to the most popular off-the-shelf CPU-based LDA method, Gensim LDAMulticore, which is fully parallelized (Rehurek and Sojka 2011). For the Gensim LDA model, we keep all default parameters, except we increase the number of passes and iterations until the coherence of the model converges for each specified number of topics. We include the final parameters used in Table A7 in the Supplementary Material. We compute the coherence measures using the Gensim CoherenceModel library by providing the 20 top words in each topic and using the default parameters for each coherence measure.

In the Supplementary Material, we also compare how our methods scale against other state-of-the-art LDA methods (see Table A4 in the Supplementary Material). In particular, we analyze LightLDA (Yuan *et al.* 2015) and WarpLDA (Chen *et al.* 2016), both of which purport tremendous increases in scale. We note neither method offers the accuracy guarantees of TLDA, nor are they actively maintained as part of a larger suite of packages (such as Gensim's LDAMulticore or Tensorly's TLDA, presented here). We note in any case that our method outperforms both in terms of time, even without the theoretical guarantees of our method (Anandkumar *et al.* 2013).[10] We also note that there are many other methods purporting to perform topic modeling at scale. As far as our review of them can determine, none purport to have the accuracy guarantees of our method, nor have they been widely adopted. For this reason, we focus our analysis on Gensim's implementation, which is, by far, the most popular and widely adopted.

As shown in Table 4, we find that the batched version of our method, running on a single GPU, achieves 10×–95× speedup over Gensim running in parallel on 79 CPU cores. The online version running on a single GPU achieves a 15×–143× speedup over Gensim run in parallel on 79 CPU cores. Notably, our method has a relatively constant convergence speed and consistent coherence for all numbers of topics. On the other hand, as the number of topics increases, the convergence time for the Gensim model increases nonlinearly and the final coherence score to which it converges decreases significantly. The runtime comparison as the number of topics increases is visualized in Figure 5. As these results indicate, our method is especially useful for analyzing larger numbers of topics, thanks to the near-constant scaling of convergence time as the number of topics increases: for more than 60 topics, our method is over 100× faster on GPU than the fully parallelized Gensim CPU LDA method.

Scaling Study We fit our TLDA method for 10, 20, 40, 60, 80, and 100 topics on four subsets of the #MeToo movement containing 1, 2, 5, and 7.97 million tweets. We keep the same vocabulary and optimal parameters from the full #MeToo timing comparison while running the model on each of these subsets. As seen in Figure 6, which displays the TLDA fit time (excluding preprocessing) for 100 topics, the convergence time for our method scales linearly with the number of documents on the GPU, while the convergence time for the Gensim LDAMulticore method increases significantly faster. Combined with our empirical finding from the full-scale #MeToo study that our method scales nearly constantly with the number of topics, this indicates that it is feasible to run both the batched and online versions of our method on even larger data. We include the plots for the remaining number of topics in Figure A2 in the Supplementary Material.

7.1.4. CPU Ablation Study

To demonstrate the benefits of our GPU implementation, we perform an ablation study comparing CPU and GPU TLDA runtime.

[10] We note that not only is our method faster than prior scalable LDA work, but it also allows for the benefits of TLDA, such as these convergence guarantees and exploiting third-order word co-occurrences, to be used for modeling large-scale data for the first time.

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

Table 4. TLDA convergence timing comparison on the full #MeToo dataset.

				Coherence			
Number of topics	Model	Time (s)	Speedup	C_v	C_{UCI}	C_{NPMI}	U_{Mass}
10	Gensim LDAMulticore	1,581.80	-	**0.421**	0.481	0.061	−4.172
	Batched TLDA	121.81	12.99	0.389	**0.489**	**0.063**	**−3.514**
	Online TLDA	**103.45**	**15.29**	0.407	0.473	0.061	−3.528
20	Gensim LDAMulticore	1,528.38	-	**0.420**	**0.493**	0.058	−4.514
	Batched TLDA	145.33	10.52	0.393	0.473	**0.062**	**−3.469**
	Online TLDA	**96.10**	**15.90**	0.370	0.440	0.056	−3.561
40	Gensim LDAMulticore	8,801.33	-	0.404	0.479	0.053	−5.075
	Batched TLDA	149.54	58.86	**0.416**	**0.499**	**0.066**	**−3.474**
	Online TLDA	**98.27**	**89.56**	0.384	0.451	0.057	−3.609
60	Gensim LDAMulticore	14,448.87	-	0.383	0.377	0.045	−5.446
	Batched TLDA	152.67	94.64	**0.402**	**0.459**	**0.060**	**−3.466**
	Online TLDA	**126.19**	**114.50**	0.380	0.433	0.055	−3.622
80	Gensim LDAMulticore	14,700.59	-	0.382	0.324	0.042	−5.434
	Batched TLDA	158.88	92.53	**0.389**	**0.465**	**0.061**	**−3.488**
	Online TLDA	**120.13**	**122.37**	0.355	0.414	0.052	−3.624
100	Gensim LDAMulticore	14,830.54	-	**0.381**	0.352	0.043	−5.176
	Batched TLDA	171.82	86.31	0.375	**0.439**	**0.058**	**−3.524**
	Online TLDA	**103.45**	**143.36**	0.355	0.417	0.052	−3.647

Note: See Rehurek and Sojka (2011) for details about Gensim LDA Multicore and Röder, Both, and Hinneburg (2015) for the coherence metrics.

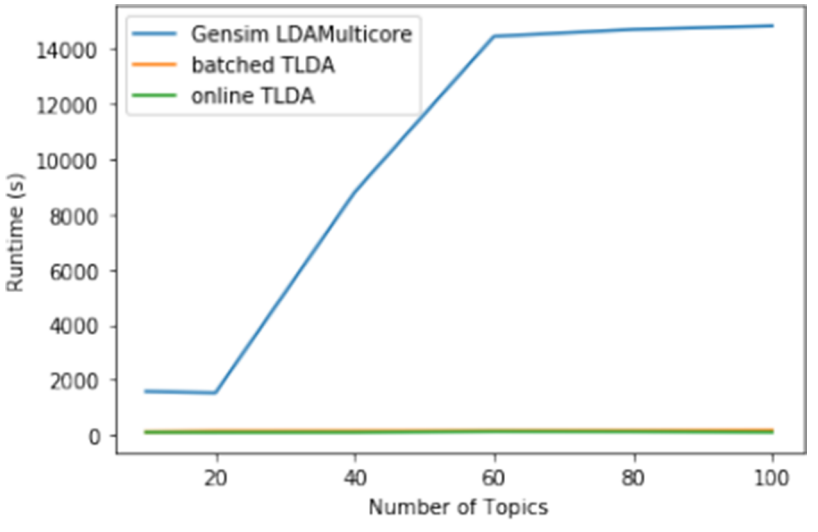


Figure 5. Runtime comparison for TLDA on GPU vs. Gensim for the full #MeToo corpus and varying numbers of topics.
Note: This shows that the runtime of our method scales nearly constantly with respect to the number of topics, while Gensim scales more than linearly.

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

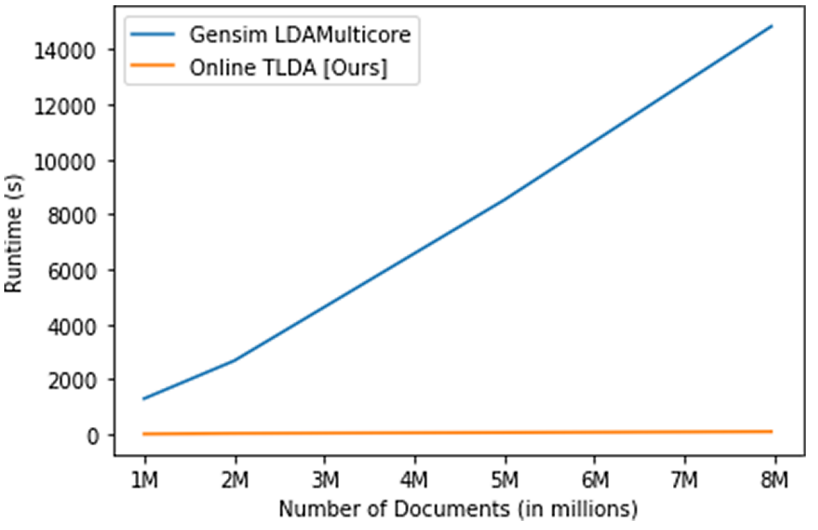


Figure 6. TLDA vs. Gensim fitting time.
Note: We compare the time to fit Gensim's LDAMulticore and our online TLDA, not including pre-processing, for 100 topics. We plot the runtime in seconds as a function of the size of the subset from the #MeToo dataset, from 1 million to 7.97 million tweets.

Table 5. Comparison of CPU and GPU runtime on the #MeToo dataset (7.97 million tweets).

Number of topics (K)	Online TLDA Time on CPU (s)	Online TLDA Time on GPU (s)	Speedup
10	1,958	103	19×
20	1,906	96	20×
40	1,937	98	20×
60	1,975	126	16×
80	1,947	120	16×
100	1,958	103	19×

As shown in Table 5 (which does not include pre-processing time), we find that the CPU version of our method has relatively constant runtime with respect to the number of topics and takes under 1 hour to converge for each number of topics on the full #MeToo dataset. Furthermore, we perform an empirical scaling study (see Figure A3 in the Supplementary Material), in which we confirm that the convergence time of our method scales linearly with the number of topics on CPU as well as on GPU. These results indicate that our method is feasible to use on CPU for datasets on the scale of tens of millions of documents for researchers who do not have access to GPU machines. However, the GPU implementation results in significant gains in TLDA convergence speed and allows for scaling to hundreds of millions and billions of documents.

7.2. Qualitative Analysis of the #MeToo Movement

Finally, we present a full-scale analysis of the evolution of two years of #MeToo Twitter discussion over time. Previous studies have shown the importance of accounting for the dynamic nature of conversation in the #MeToo movement (Liu *et al.* 2019), and here we analyze the topical development of tweets concerning #MeToo from just before the start of the movement in August 2017 through September 2019. Understanding the topical development of the #MeToo movement would be important for scholars who wish to test theories about how social movements organize online, for studies of feminism or gender politics, for two examples.

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

To develop this dynamic analysis, we iteratively grow the corpus of tweets and estimate the TLDA model. That is, first, we fit the model on August and September 2017 data to capture the discussion immediately preceding the time when #MeToo discussion went viral on Twitter. Then, we add the next month and estimate the entire model again. We repeat this process for each subsequent month until we reach the end of the dataset. In Figure 1, we report the topic for selected months with the largest weight α_i for both pro- and counter-#MeToo topics. We call these topics the most prominent. The relative size of the labels indicates the relative prominence of the respective topics. In Figure 2, we report the political topic with the largest α_i, which we call the most politically salient.

This approach leads to three key qualitative findings: first, topics related to politically salient news events are generally ephemeral, changing often as we grow the dataset over time. In contrast, topical prominence related to personal testimonies, coordinating protests, and supporting other participants in the #MeToo movement is persistent over time. Third, discussion around counter-#MeToo topics declines in prominence over time and is subsumed into one topic by September 2019. These findings suggest avenues for the study of the temporal evolution of political and social coordination in mass movements, especially on social media, as most research using this type of data to study social movements has not examined the longer-term dynamics of these forms of political engagement (Jost *et al.* 2018; Steinert-Threlkeld 2017). Moreover, these results indicate that dynamic changes in topic evolution on social media may be in response to changing news events, another area ripe for new research using large social media and news media datasets using methods like TLDA.

7.3. Election 2020: Elite Political Communication and the Losers' Effect

In this second application, we look at another source of real-time social media data relevant to how political elites communicate with and coordinate their supporters, especially in the face of electoral defeat. A peaceful transfer of power is a necessary indicator of a healthy democracy. Studies into how political elites respond to electoral defeat offer critical insights into how democracies persist, even in polarized or politically fraught periods of history. In light of the 2020 election, longstanding questions around how politicians organize their supporters around election were especially salient. Previous studies into political anger and the losers' effect have relied on survey-based evidence, and produced valuable insights into voter psychology and political behavior (Craig *et al.* 2006; Sinclair, Smith, and Tucker 2018); however, these methods are static and rely on recalled emotions after an election. Building on this important work, our methodology allows researchers to study how a politician coordinates with his or her politically engaged supporters online. With this type of data, we can study online political behaviors in real-time—President Trump was active online and especially on Twitter (Li *et al.* 2023), so these conditions allow us to better understand how electoral losers respond to electoral defeat. For policy makers, such methods might allow for real-time detection of the spread of effective attempts to inform the public, while highlighting messaging that angers, rather than informs, the voting public.

As with #MeToo, this data was collected based on a keyword search employing the social media data collection techniques in Cao, Adams-Cohen, and Alvarez (2021) and Li *et al.* (2023). The data are lightly structured, so our method can offer insights into the latent structure of this politically engaged demographic: online supporters. In this case, a comprehensive collection of tweets related to keywords related to the administration of the 2020 Presidential election served as the basis of the data collection. Overall, the data are composed of 29,711,862 tweets collected from September 1, 2020 through the Inauguration on January 20, 2021. These data were collected in real time as the tweets were being posted. In large part thanks to this real-time collection of data, we were able to capture many tweets prior to their being deleted or moderated, giving us an unfiltered look at social media activity during a critical period in American electoral politics.

In Figure 7, we report the daily number of tweets collected by our keyword search. The data are most voluminous on November 4th, the day of the election, with nearly 3 million election-related tweets. There is a second peak of just under 1.5 million tweets on November 7th, the day the media outlets declared Joe Biden the winner of the election. Activity remains high following the election with nearly 250 thousand to 500 thousand tweets following the election.

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

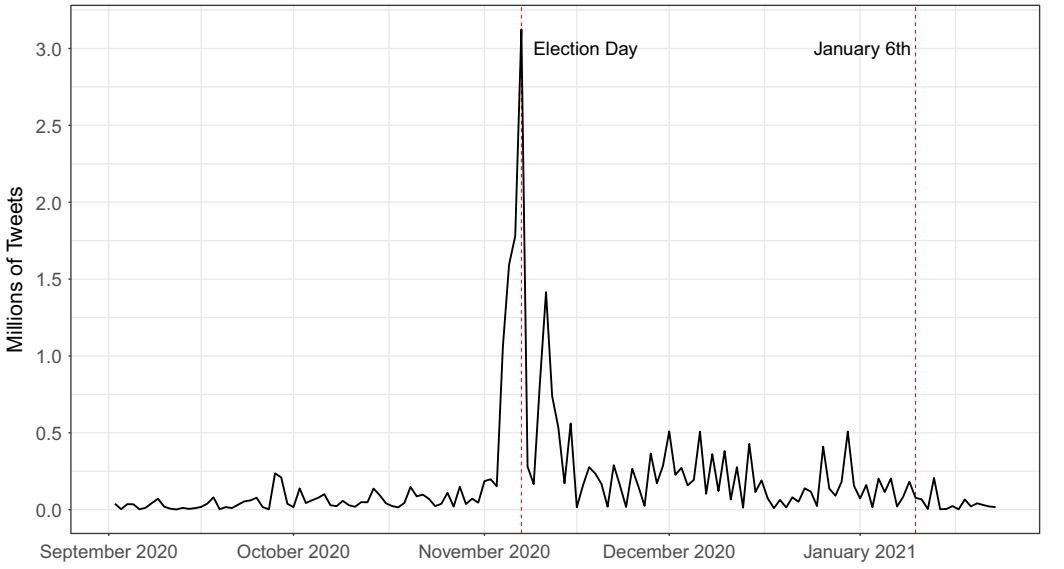


Figure 7. Number of tweets per day.

The TLDA framework offers potential new insights as a data discovery method for a critically important area of political science—how the public and politicians organize messaging surrounding the legitimacy of elections on social media. Large-scale social media data concerning the legitimacy of the election are particularly suited for the TLDA framework because discerning meaningful structure from these data would be impractical without automated methods. More than merely large, the data are unstructured because they are directly collected in real-time from people's authentic online thoughts. In terms of generating new theories about the role of online behavior in disseminating information about the election amplifying the loser's effect, actively organizing political supporters, and its direct role in helping to coordinate the rally and riots on January 6th, a data discovery method like TLDA is the most computationally feasible way to study these data real-time and at scale.

7.4. Qualitative Analysis of Trump-Related Social Media Activity in the 2020 Election

We now report an analysis based on the outputs of our TLDA analysis—we hope this serves as an example of how political scientists might employ our framework to engage in data discovery for unstructured data at large scale. We report descriptive findings on a full population of tweets covering a topic of critical importance to students of political behavior and electoral politics. To that end, we recover 30 topics overall. For this analysis, we classify the topics into three main categories based on the following criteria: of the discovered topics, we find three broadly related categories: discrediting the election, discussing legal challenges to the election, and tweets verifying the election result.

Notably, in Figure 8, we observe relative stability in the daily share of tweets belonging to each category both across time and between the relative share of each of the categories. We see that tweets discrediting the election account for 25% of all tweets on average, with a relative peak on January 6th. We see that in the days after January 6th, the share of tweets related to verifying the election results actually drops.

7.4.1. Pre-Processing

Following the same pre-processing as for the #MeToo data (i.e., we removed all words that appear in fewer than 0.002% of documents and in more than 50% of documents), we generate a vocabulary of

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

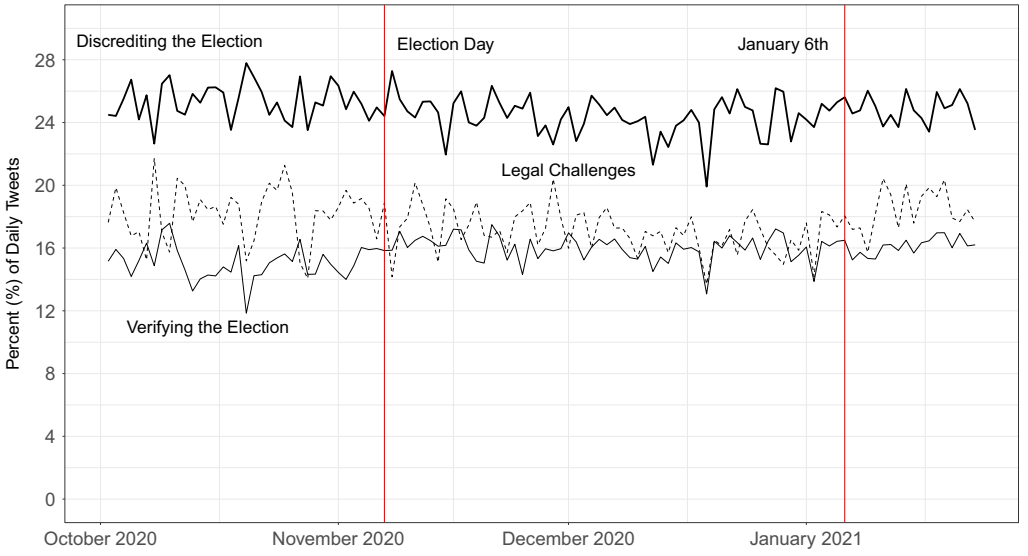


Figure 8. Topical composition over time: In this figure, we report the average share of tweets belonging to one of three main categories of topics with a greater than 90% probability.

3,788 words. As in the case of the #MeToo analysis, we perform all pre-processing using the Rapids GPU Framework. We note that preprocessing the entire dataset takes only 3,661 seconds (or 1.02 hours).

7.4.2. Timing

In addition, to demonstrate that our method can scale to billions of documents, we use a large COVID dataset to simulate a dataset of 1,043,009,932 documents. To do so, we loop through the entirety of the data four times and update the online version of the method on each batch once within each iteration. We report full results in the Supplementary Material.

In Table 1, we show that the model converges in 3.47 hours. We include the top 20 words from each topic and word clouds for the top 3 topics in Section 1.5 of the Supplementary Material to illustrate that our model produces coherent topics. We also find that our method scales linearly to the simulated data, as it takes 13.16 hours to analyze 1.04 billion documents. This illustrates the feasibility of using our method for tuning, estimating, and analyzing topic models on social media data at a large scale.

8. Conclusion

Applied researchers have access to larger and larger datasets of text information. These datasets may come from social media platforms like Twitter, Reddit, or Facebook (Steinert-Threlkeld 2018), congressional speeches, news media archives, legislative text, campaign websites, and senatorial memos. These data sources are being used for critical new studies in political science, across a variety of topics, including political behavior, political opinion, protest movements, legislative agendas, and collective action. Other large and complex text datasets are being compiled and used in the humanities (Dennis-Henderson 2020). The widespread availability of unique and high-dimensional text data is opening new doors for applied researchers in the field.

However, off-the-shelf methods, such as LDA, are computationally inefficient and unusable for analyzing large, high-dimensional text datasets. Our approach provides a feasible scale and a method with theoretical foundations for its statistical properties. This compares favorably, as we noted earlier, with proprietary LLMs. New LLM methods show increasing promise, but yet pose lingering challenges for academic researchers, including a lack of strong statistical theoretical foundations. We hope in future

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

research to show that these statistical properties serve as a foundation for further validation of the quantities of interest researchers aim to study.

In this article, we proposed an efficient and theoretically-founded TLDA approach for estimating topic models, which we have shown can be fitted on large text datasets many orders of magnitude faster than existing off-the-shelf topic modeling methods. Our implementation of the TLDA method builds on the TensorLy library. We aim to democratize the analysis of large-text datasets by offering a method that scales on both GPU and CPU, is fully open-source, and whose mathematical underpinnings are clearly communicated. We hope the wide availability of this method will enable new lines of research on previously out-of-reach scales.

Acknowledgments. The authors thank Ryan P. Kennedy, Patrick T. Brandt, and the participants at the 2024 "TexMeth" Workshop for their comments and suggestions. Some of S.K.'s work on this project was supported by Summer Undergraduate Research Fellowships from Caltech. We thank the Anima AI + Science Lab and the Alvarez Lab for their comments and feedback.

Data Availability Statement. Replication code for this article is available via Dataverse at Ebanks *et al.* (2025) at https://doi.org/10.7910/DVN/OKPRJG.

Author Contributions. Conceptualization: S.K., D.E., J.K., A.L., R.M.A., and A.A.; Data collection and curation: S.K., D.E., and R.M.A.; Formal analysis: S.K., D.E., J.K., A.L., and A.A.; Funding acquisition: R.M.A. and A.A.; Methodology: S.K., D.E., J.K., A.L., R.M.A., and A.A.; Project administration: R.M.A. and A.A.; Software: S.K., D.E., and J.K.; Writing and editing: S.K., D.E., J.K., A.L., R.M.A., and A.A.

Supplementary Material. The supplementary material for this article can be found at https://doi.org/10.1017/pan.2025.10024.

References

Anandkumar, A., D. P. Foster, D. J. Hsu, S. M. Kakade, and Y.-K. Liu. 2012. "Two SVDs Suffice: Spectral Decompositions for Probabilistic Topic Modeling and Latent Dirichlet Allocation." Preprint, arXiv:1204.6703. http://arxiv.org/abs/1204.6703.

Anandkumar, A., D. P. Foster, D. J. Hsu, S. M. Kakade, and Y.-K. Liu. 2013. "A Spectral Algorithm for Latent Dirichlet Allocation." In *Advances in Neural Information Processing Systems*, edited by F. Pereira and C.J. Burges and L. Bottou and K.Q. Weinberger, vol. 25. Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2012/file/15d4e891d784977cacbfcbb00c48f133-Paper.pdf

Anandkumar, A., R. Ge, D. Hsu, S. M. Kakade, and M. Telgarsky. 2014. "Tensor Decompositions for Learning Latent Variable Models." *Journal of Machine Learning Research* 15 (80): 2773–2832. http://jmlr.org/papers/v15/anandkumar14b.html

Barberá, P. 2015. "Birds of the Same Feather Tweet Together: Bayesian Ideal Point Estimation Using Twitter Data." *Political Analysis* 23 (1): 76–91.

Barberá, P., A. E. Boydstun, S. Linn, R. McMahon, and J. Nagler. 2021. "Automated Text Classification of News Articles: A Practical Guide." *Political Analysis* 29 (1): 19–42. https://doi.org/10.1017/pan.2020.8

Barberá, P., et al. 2019. "Who Leads? Who Follows? Measuring Issue Attention and Agenda Setting by Legislators and the Mass Public Using Social Media Data." *American Political Science Review* 113 (4): 883–901.

Barberá, P., J. T. Jost, J. Nagler, J. A. Tucker, and R. Bonneau. 2015. "Tweeting from Left to Right: Is Online Political Communication More than an Echo Chamber?" *Psychological Science* 26 (10): 1531–1542.

Bartl, M., M. Nissim, and A. Gatt. 2020. "Unmasking Contextual Stereotypes: Measuring and Mitigating BERT's Gender Bias." In *Proceedings of the Second Workshop on Gender Bias in Natural Language Processing*, edited by M. R. Costa-jussà, C. Hardmeier, W. Radford, and K. Webster, 1–6. Barcelona, Spain: Association for Computational Linguistics. https://aclanthology.org/2020.gebnlp-1.1/

Benoit, K., K. Munger, and A. Spirling. 2019. "Measuring and Explaining Political Sophistication through Textual Complexity." *American Journal of Political Science* 63 (2): 491–508.

Blei, D. M., A. Y. Ng, and M. I. Jordan. 2003. "Latent Dirichlet Allocation." *Journal of Machine Learning Research* 3: 993–1022.

Brown, D. 2018. "19 Million Tweets Later: A Look at #MeToo a Year after the Hashtag Went Viral." *USA Today.*

Cao, J., N. Adams-Cohen, and R. Michael Alvarez. 2021. "Reliable and Efficient Long-Term Social Media Monitoring." *Journal of Computer and Communications* 9: 97–109. https://doi.org/10.4236/jcc.2021.910006

Chen, J., K. Li, J. Zhu, and W. Chen. 2016. "WarpLDA: A Cache Efficient O(1) Algorithm for Latent Dirichlet Allocation." *Proceedings of the VLDB Endowment* 9 (10): 744–755. https://doi.org/10.14778/2977797.2977801

Chong, D. 1991. *Collective Action and the Civil Rights Movement*. Chicago, IL: University of Chicago Press.

Clark-Parsons, R. 2022. *Networked Feminism how Digital Media Makers Transformed Gender Justice Movements*. Oakland, CA: University of California Press.

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

Craig, S. C., M. D. Martinez, J. Gainous, and J. G. Kane. 2006. "Winners, Losers, and Election Context: Voter Responses to the 2000 Presidential Election." *Political Research Quarterly* 59 (4): 579–592. https://doi.org/10.1177/106591290605900407

Dennis-Henderson, A. G. 2020. "*Analysis of World War One Diaries Using Natural Language Processing*." PhD diss., The University of Adelaide. https://hdl.handle.net/2440/129622

Denny, M. J., and A. Spirling. 2018. "Text Preprocessing for Unsupervised Learning: Why it Matters, when it Misleads, and What to Do about it." *Political Analysis* 26 (2): 168–189.

Dimitrov, D., et al. 2020. "TweetsCOV19—A Knowledge Base of Semantically Annotated Tweets about the COVID-19 Pandemic." In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management CIKM '20 Virtual Event*, 2991–2998. Ireland: Association for Computing Machinery. https://doi.org/10.1145/3340531.3412765

Ebanks, D., A. Anandkumar, J. Kossaifi, K. Sara, and R. Alvarez. 2025. "Replication Data for: Analyzing Political Text at Scale with Online Tensor LDA." https://doi.org/10.7910/DVN/OKPRJG

Friedersdorf, C. 2017. "The Significance of Millions in the Streets." https://web.archive.org/web/20170126005530/https://www.theatlantic.com/politics/archive/2017/01/the-significance-of-millions-in-the-streets/514091/

Gilardi, F., M. Kubli, T. Gessler, and S. Müller. 2022. "Social Media and Political Agenda Setting." *Political Communication* 39 (1): 39–60.

Grimmer, J., and G. King. 2011. "General Purpose Computer-Assisted Clustering and Conceptualization." *Proceedings of the National Academy of Sciences* 108 (7): 2643–2650. https://doi.org/10.1073/pnas.1018067108

Grimmer, J., M. E. Roberts, and B. M. Stewart. 2022. *Text as Data: A New Framework for Machine Learning and the Social Sciences*. Princeton, NJ: Princeton University Press.

Grimmer, J., and B. M. Stewart. 2013. "Text as Data: The Promise and Pitfalls of Automatic Content Analysis Methods for Political Texts." *Political Analysis* 21 (3): 267–297. http://www.jstor.org/stable/24572662

Hannak, A., E. Anderson, L. F. Barrett, S. Lehmann, A. Mislove, and M. Riedewald. 2021. "Tweetin' in the Rain: Exploring Societal-Scale Effects of Weather on Mood." *Proceedings of the International AAAI Conference on Web and Social Media* 6 (1): 479–482. https://ojs.aaai.org/index.php/ICWSM/article/view/14322

Hoffman, M., F. Bach, and D. Blei. 2010. "Online Learning for Latent Dirichlet Allocation." In *Advances in Neural Information Processing Systems*, edited by J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, vol. 23. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2010/file/71f6278d140af599e06ad9bf1ba03cb0-Paper.pdf

Huang, F., U. N. Niranjan, M. U. Hakeem, and A. Anandkumar. 2015. "Online Tensor Methods for Learning Latent Variable Models." *Journal of Machine Learning Research* 16 (86): 2797–2835. http://jmlr.org/papers/v16/huang15a.html

Janzamin, M., R. Ge, J. Kossaifi, and A. Anandkumar. 2019. "Spectral Learning on Matrices and Tensors." *Foundations and Trends® in Machine Learning* 12 (5–6): 393–536. https://doi.org/10.1561/2200000057

Jost, J. T., et al. 2018. "How Social Media Facilitates Political Protest: Information, Motivation, and Social Networks." *Political Psychology* 39 (S1): 85–118. https://doi.org/10.1111/pops.12478.

Kann, C., S. Hashash, Z. Steinert-Threlkeld, and R. Michael Alvarez. 2023. "Collective Identity in Collective Action: Evidence from the 2020 Summer BLM Protests." *Frontiers in Political Science* 5. https://doi.org/10.3389/fpos.2023.1185633.

King, G., and D. Hopkins. 2010. "Extracting Systematic Social Science Meaning from Text." *American Journal of Political Science* 54 (1): 229–247.

Kocielnik, R., S. Prabhumoye, V. Zhang, R. Jiang, R. M. Alvarez, and A. Anandkumar. 2023. "BiasTestGPT: Using ChatGPT for Social Bias Testing of Language Models." Preprint, arXiv:2302.07371.

Kolda, T. G., and B. W. Bader. 2009. "Tensor Decompositions and Applications." *SIAM Review* 51 (3): 455–500.

Kossaifi, J., Y. Panagakis, A. Anandkumar, and M. Pantic. 2019. "TensorLy: Tensor Learning in Python." *Journal of Machine Learning Research* 20 (26): 1–6. http://jmlr.org/papers/v20/18-277.html.

Lauderdale, B. E., and T. S. Clark. 2014. "Scaling Politically Meaningful Dimensions Using Texts and Votes." *American Journal of Political Science* 58 (3): 754–771.

Larson, J. M., J. Nagler, J. Ronen, and J. A. Tucker. 2019. "Social Networks and Protest Participation: Evidence from 130 Million Twitter Users." *American Journal of Political Science* 63 (3): 690–705.

Li, Z., J. Cao, N. Adams-Cohen, and R. Michael Alvarez. 2023. "The Effect of Misinformation Intervention: Evidence from Trump's Tweets and the 2020 Election." In *Disinformation in Open Online Media*, edited by D. Ceolin, T. Caselli, and M. Tulin, 88–102. Cham: Springer Nature Switzerland.

Linegar, M., R. Kocielnik, and R. Michael Alvarez. 2023. "Large Language Models and Political Science." *Frontiers in Political Science* 5. https://doi.org/10.3389/fpos.2023.1257092.

Liu, A., M. Srikanth, N. Adams-Cohen, R. M. Alvarez, and A. Anandkumar. 2019. "Finding Social Media Trolls: Dynamic Keyword Selection Methods for Rapidly-Evolving Online Debates." Preprint, arXiv:1911.05332.

Lucas, C., R. A. Nielsen, M. E. Roberts, B. M. Stewart, A. Storer, and D. Tingley. 2015. "Computer-Assisted Text Analysis for Comparative Politics." *Political Analysis* 23 (2): 254–277. https://doi.org/10.1093/pan/mpu019

Metzger, M. M. D., R. Bonneau, J. Nagler, and J. A. Tucker. 2016. "Tweeting Identity? Ukrainian, Russian, and #Euromaidan." *Journal of Comparative Economics* 44 (1): 16–40.

Monroe, B. L., M. P. Colaresi, and K. M. Quinn. 2017. "Fightin' Words: Lexical Feature Selection and Evaluation for Identifying the Content of Political Conflict." *Political Analysis* 16 (4): 372–403. https://doi.org/10.1093/pan/mpn018

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press

Munger, K. 2017. "Tweetment Effects on the Tweeted: Experimentally Reducing Racist Harassment." *Political Behavior* 39: 629–649. https://doi.org/10.1007/s11109-016-9373-5

Nozza, D., F. Bianchi, and D. Hovy. 2021. "HONEST: Measuring Hurtful Sentence Completion in Language Models." In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, edited by K. Toutanova, et al., 2398–2406. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.naacl-main.191

Olson Jr., M. 1971. *The Logic of Collective Action: Public Goods and the Theory of Groups, with a New Preface and Appendix*, vol. 124. Cambridge, MA: Harvard University Press.

Papalexakis, E. E., C. Faloutsos, and N. D. Sidiropoulos. 2016. "Tensors for Data Mining and Data Fusion: Models, Applications, and Scalable Algorithms." *ACM Transactions on Intelligent Systems and Technology* 8 (2): 1–44.

RAPIDS Development Team. 2018. "RAPIDS: Collection of Libraries for End to End GPU Data Science." Nvidia. https://rapids.ai

Rehurek, R., and P. Sojka. 2011. "Gensim–Python Framework for Vector Space Modelling." *NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic* 3 (2).

Roberts, M. E., et al. 2014. "Structural Topic Models for Open-Ended Survey Responses." *American Journal of Political Science* 58 (4): 1064–1082.

Roberts, M. E., B. M. Stewart, and D. Tingley. 2016. "Navigating the Local Modes of Big Data." In *Computational Social Science*, edited by R. M. Alvarez. Cambridge, UK: Cambridge University Press Cambridge.

Röder, M., A. Both, and A. Hinneburg. 2015. "Exploring the Space of Topic Coherence Measures." In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining WSDM '15*, 399–408. Shanghai: Association for Computing Machinery. https://doi.org/10.1145/2684822.2685324

Sheng, E., K.-W. Chang, P. Natarajan, and N. Peng. 2019. "The Woman Worked as a Babysitter: On Biases in Language Generation." 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)*, edited by K. Inui, J. Jiang, V. Ng, and X. Wan, 3407–3412. Hong Kong: Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-1339

Sidiropoulos, N. D., L. De Lathauwer, X. Fu, K. Huang, E. E . Papalexakis, and C. Faloutsos. 2017. "Tensor Decomposition for Signal Processing and Machine Learning." *Transactions Signal Processing* 65 (13): 3551–3582.

Sinclair, B., S. S. Smith, and P. D. Tucker. 2018. ""It's Largely a Rigged System": Voter Confidence and the Winner Effect in 2016." *Political Research Quarterly* 71 (4): 854–868.

Sinnenberg, L., et al. 2016. "Twitter as a Potential Data Source for Cardiovascular Disease Research." *JAMA Cardiology* 1 (9): 1032–1036. https://doi.org/10.1001/jamacardio.2016.3029

Steinert-Threlkeld, Z. 2018. *Twitter as Data*. Cambridge, UK: Cambridge University Press.

Steinert-Threlkeld, Z. C. 2017. "Spontaneous Collective Action: Peripheral Mobilization during the Arab Spring." *American Political Science Review* 111 (2): 379–403. https://doi.org/10.1017/S0003055416000769

Swierczewski, C., S. Bodapati, A. Beniwal, D. Leen, and A. Anandkumar. 2019. "Large Scale Cloud Deployment of Spectral Topic Modeling." https://parlearning.github.io/papers%5C_2019/ParLearning%5C_2019%5C_04.pdf

Tillery, A. B. 2019. "What Kind of Movement Is Black Lives Matter? The View from Twitter." *The Journal of Race, Ethnicity, and Politics* 4 (2): 297–323. https://doi.org/10.1017/rep.2019.17

Wang, S., H. Liu, A. Gaihre, and H. Yu. 2023. "EZLDA: Efficient and Scalable LDA on GPUs." in IEEE Access, vol. 11, pp. 100165–100179, https://doi.org/10.1109/ACCESS.2023.3315239.

Xie, X., Y. Liang, X. Li, and W. Tan. 2019. "CuLDA: Solving Large-Scale LDA Problems on GPUs." In *Proceedings of the 28th International Symposium on High-Performance Parallel and Distributed Computing HPDC '19*, 195–205. Phoenix, AZ: Association for Computing Machinery. https://doi.org/10.1145/3307681.3325407

Yu, H.-F., C.-J. Hsieh, H. Yun, S. V. N. Vishwanathan, and I. S. Dhillon. 2015. "A Scalable Asynchronous Distributed Algorithm for Topic Modeling." In *Proceedings of the 24th International Conference on World Wide Web WWW '15*, 1340–1350. Florence: International World Wide Web Conferences Steering Committee. https://doi.org/10.1145/2736277.2741682

Yuan, J., et al. 2015. "LightLDA: Big Topic Models on Modest Computer Clusters." In *Proceedings of the 24th International Conference on World Wide Web WWW '15*, 1351–1361. Florence: International World Wide Web Conferences Steering Committee. https://doi.org/10.1145/2736277.2741115

Zhao, J., T. Wang, M. Yatskar, V. Ordonez, and K.-W. Chang. 2018. "Gender Bias in Coreference Resolution: Evaluation and Debiasing Methods." Preprint, arXiv:1804.06876.

Cite this article: Kangaslahti, S., Ebanks, D., Kossaifi, J., Liu, A., Alvarez, R. M. and Anandkumar, A. (2026). Analyzing Political Text at Scale with Online Tensor LDA. *Political Analysis*, 53–77. https://doi.org/10.1017/pan.2025.10024

https://doi.org/10.1017/pan.2025.10024 Published online by Cambridge University Press