Analysis of Fake News Using Topic Modeling

Murad Bashirov (CS), Yosep Shin (EE), Taeyoon Woo (MS)

KAIST

Fall 2022

Abstract

In this research, we analyzed the topics of fake news using topic modeling. We used a dataset from Kaggle.com, containing 40000 fake and real news. We used LDA modeling and were able to conclude that fake news has more specific topics, compared to real news having more general topics. We also suggest a few methods to improve and analyze further.

I. Introduction

At first, our research question was about "how fake news affects public opinion?". However, it was a quite challenging task to identify the relationship between them. So, we focused on the features of fake news, rather than the effects. We thought about what the characteristics of fake news alone would be, compared to real news, and came up with the following question. "Fake news was written with the intention of attacking a specific target, so they would cover only the negative parts of the object. Therefore, the topic they cover will be limited compared to real news." For example, let's say a politician came up with a policy that was good from an economic point of view, but not from an environmental point of view. What we can expect is that fake news focuses only on the keyword "environment destruction" and amplifies politician's faults, while "economic growth and environment destruction" will appear simultaneously in news articles written from a fair perspective.

Hence, we tried to answer the following research questions. "Is the topic of fake news limited compared to real news? If it's true, Why does it happen? Is it because fake news only covers some specific, negative topics trying to insult someone?"

II. Background

Structural Topic Modeling, or STM, is a way to find topics of a document using a machine learning model. Specifically, there is a famous model called LDA that stands for Latent Dirichlet Allocation(Blei et al. (2003)). It takes the corpus of documents as input, with the number of topics as a hyperparameter, then returns topics. Topics are represented by the shape of lists of words, attached with probability. Those words describe each topic, for example, if a topic contains the word "Trump" with a probability of 0.027, it means a document classified in this topic has a 0.027 chance of having the word "Trump".

But to generate topics, LDA requires some input parameters, such as the number of topics n, and two prior beliefs: a prior belief on document-topic distribution, and a prior belief on topic-word distribution. The optimal values for these priors can be found from evaluating LDA models.

In order to evaluate LDA models, one can use coherence values described in the Röder et al. (2015). Specifically, we used the C_v value, which is well explained from a high-level perspective in the Röder (2015)

> C_v is based on a sliding window, a one-set segmentation of the top words and an indirect confirmation measure that uses NPMI (normalized PMI, proposed in Aletras and Stevenson (2013)) and the cosines similarity.
>
> This coherence measure retrieves cooccurrence counts for the given words using a sliding window and the window size 110. The counts are used to calculate the NPMI of every top word to every other top word, thus, resulting in a set of vectors—one for every top word. The one-set segmentation of the top words leads to the calculation of the similarity between every top word vector and the sum of all top word vectors. As a similarity measure, the cosines are used. The coherence is the arithmetic mean of these similarities. (Note that this was the best coherence measure in our evaluation.)

Another way of evaluating topic models is by calculating the Jaccard Similarity, Jaccard (1912), which is obtained by dividing the size of intersection of two sets over their size of union:

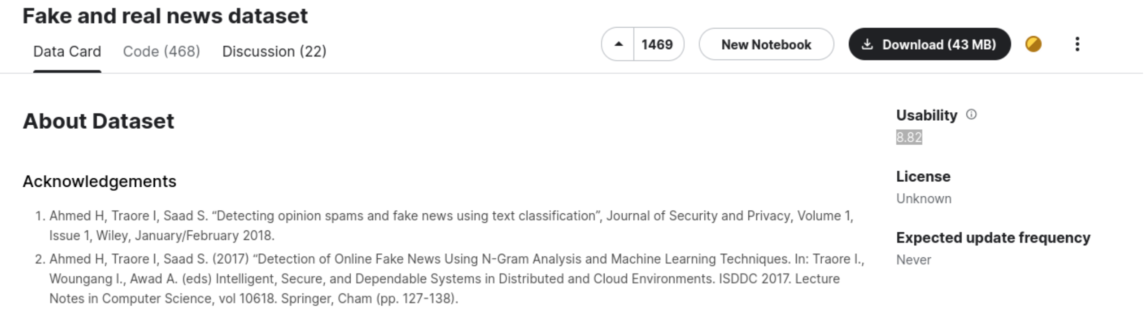
$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

When the value of Jaccard similarity is lower, then two sets are more distinct. Su et al. (2016) and Greene et al. (2014) determine topic similarities with an average Jaccard coefficient. Average Jaccard calculates the average of the Jaccard scores between every pair of subsets in two lists. We can use this metric to determine mean stability between a topic and next topic.

III. Data and Methods

i. Dataset

The dataset we used for this study was from the papers Ahmed et al. (2017) and Ahmed et al. (2018). We obtained the dataset from `https://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-news-dataset`. The dataset contained two CSV files: Fake.csv and True.csv. The fake news dataset had 23481, and real news had 21417 articles. The fake news were collected from



Fake and real news dataset

About Dataset

Acknowledgements

1. Ahmed H, Traore I, Saad S. “Detecting opinion spams and fake news using text classification”, Journal of Security and Privacy, Volume 1, Issue 1, Wiley, January/February 2018.
2. Ahmed H, Traore I, Saad S. (2017) “Detection of Online Fake News Using N-Gram Analysis and Machine Learning Techniques. In: Traore I., Woungang I., Awad A. (eds) Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments. ISDDC 2017. Lecture Notes in Computer Science, vol 10618. Springer, Cham (pp. 127-138).

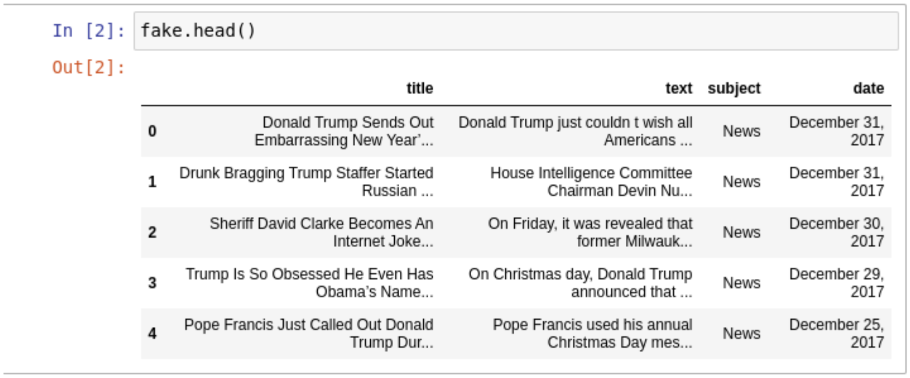
Usability
8.82

License
Unknown

Expected update frequency
Never

Figure 1: *Obtaining dataset from Kaggle*

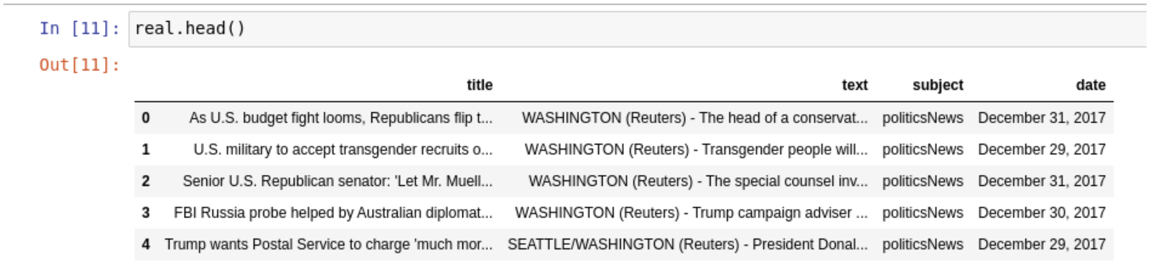
different sources. They were collected from unreliable websites that are marked by Politifact (a fact-checking organization in the USA) and Wikipedia. The real news were collected by crawling articles on Reuters.com, which is a reliable news website. Each entry in the dataset contained the following information: title, text, and date they were published. However, in this study, we mainly focused on articles' text.



```
In [2]: fake.head()
Out[2]:
```

	title	text	subject	date
0	Donald Trump Sends Out Embarrassing New Year'...	Donald Trump just couldn t wish all Americans ...	News	December 31, 2017
1	Drunk Bragging Trump Staffer Started Russian ...	House Intelligence Committee Chairman Devin Nu...	News	December 31, 2017
2	Sheriff David Clarke Becomes An Internet Joke...	On Friday, it was revealed that former Milwauk...	News	December 30, 2017
3	Trump Is So Obsessed He Even Has Obama's Name...	On Christmas day, Donald Trump announced that ...	News	December 29, 2017
4	Pope Francis Just Called Out Donald Trump Dur...	Pope Francis used his annual Christmas Day mes...	News	December 25, 2017

Figure 2: *Fake news dataset contents*



```
In [11]: real.head()
Out[11]:
```

	title	text	subject	date
0	As U.S. budget fight looms, Republicans flip t...	WASHINGTON (Reuters) - The head of a conservat...	politicsNews	December 31, 2017
1	U.S. military to accept transgender recruits o...	WASHINGTON (Reuters) - Transgender people will...	politicsNews	December 29, 2017
2	Senior U.S. Republican senator: 'Let Mr. Muell...	WASHINGTON (Reuters) - The special counsel inv...	politicsNews	December 31, 2017
3	FBI Russia probe helped by Australian diplomat...	WASHINGTON (Reuters) - Trump campaign adviser ...	politicsNews	December 30, 2017
4	Trump wants Postal Service to charge 'much mor...	SEATTLE/WASHINGTON (Reuters) - President Donal...	politicsNews	December 29, 2017

Figure 3: *Fake news dataset contents*

ii. Methods

ii.1 Overview

Our approach can be summarized by three steps:

1. Data preprocessing
2. LDA models training
3. Metric analysis

ii.2 Data preprocessing

In order to classify topics using LDA, first we need to prepare the corpus with the preprocessed words. For this step, we first removed any punctuations from the texts. Then we split sentences into words, made them lowercase, and ignored words that were too short or too long (minimum

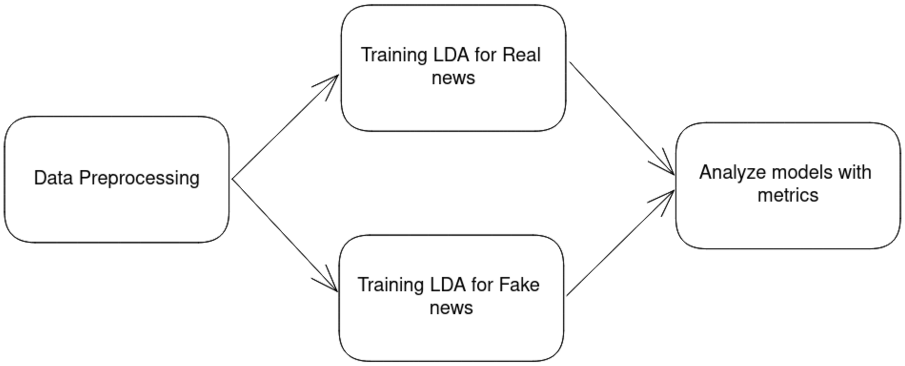
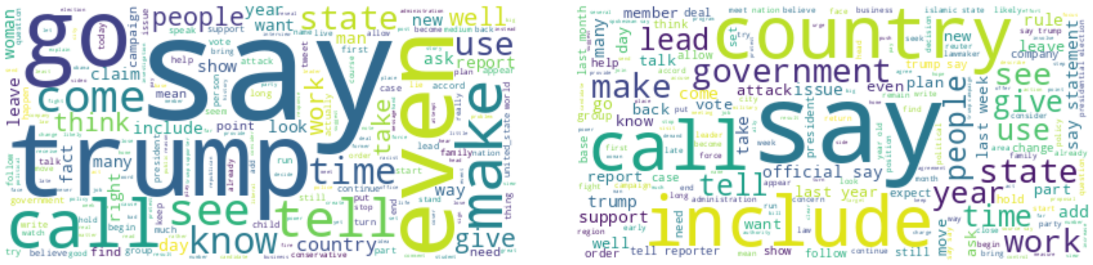

Data Preprocessing
Training LDA for Real news
Training LDA for Fake news
Analyze models with metrics

Figure 4: *Method Pipeline*

length 2, maximum length 15). Then, we removed stop words from the text. Stop words are words such that they don't contribute anything to the main idea of the text, and they have really high occurrence in any text, so they may affect the topic model significantly. In order to remove stop words, we used open source python library Spacy from Honnibal et al. (2020). After removing stop words, we lemmatized the words using Spacy again. After that, we made bigrams and added them to the corpus. Bigrams are the pair of words that appear frequently throughout the dataset. Examples from our dataset can be "police_brutality" or "north_korea".



(a) *Fake news*

(b) *Real news*

Figure 5: *Word Clouds of preprocessed datasets*

ii.3 Model Training

We used the python library gensim from Rehurek and Sojka (2011) for the implementation of LDA. Specifically, we used `LdaMulticore` model with 15 workers in order to speed up the training process.

As mentioned in the background section, apart from the corpus of documents and number of topics, the LDA model also takes two hyperparameters. In order to find those hyperparameters, we trained the model with a fixed number of topics, and changing parameters. In the end, the optimal parameters came out as symmetric (fixed symmetric prior of 1 / num_topics) for document-topic and 0.01 for word-topic distribution. The next step was training the models with different number of topics. In machine learning, when one trains a model when the number of iterations for topic (epochs, passes) increases, the results usually came out better at till some point. However, increasing epochs also increases training time. So we weren't able to train the model with a high number of epochs, 1 pass for fake news and 7 passes for real news were enough to get meaningful results. The models were trained on AMD Ryzen 7 5800H CPU, and training time took over 2 days. We trained the models for fake news, ranging from 2 to 50 and for real news, ranging from 2 to 130.

ii.4 Analyze metrics

After training was done, we calculated coherence scores for each model using gensim's `CoherenceModel`. For Jaccard similarity, we simply used basic python to find the length of intersection and union of sets and the mean of those similarities. For an optimal number of topics we want the coherence value to be maximum and similarity to be minimum, or in other words, the difference between coherence and similarity value should be maximum, with similarity being less than coherence.

IV. Results

After analysis, we found that the optimal number of fake news is 26, and for real news it's 100. From the graph of fake news it seems like we got good results, because the topic overlap reaches minimum and at that time coherence reaches maximum. But for real news the lines are somewhat parallel, we can interpret this as the actual optimal number of topics is not in the range of 2-130, but more, which we can interpret that the real news are more general than fake news. Here are example topics from fake news with 26 number of topics:

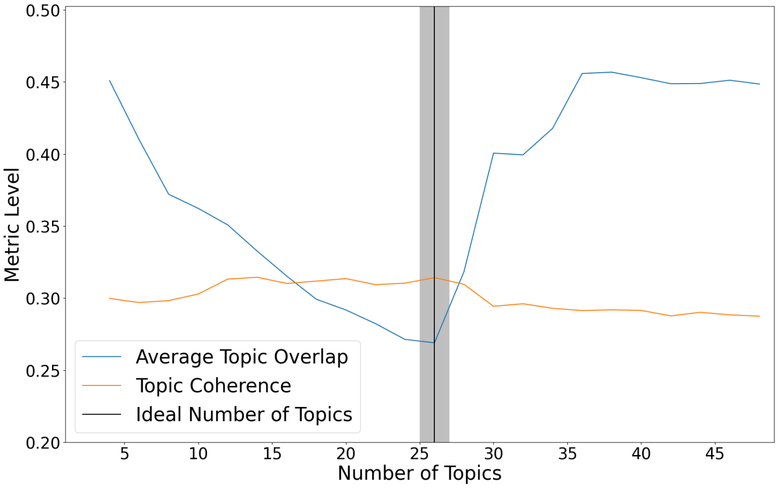


Figure 6: *Fake news metrics*

- police, say, black, officer, man, people
- trump, say, go, campaign, make
- refugee, year, make, rape
- student, school, medium
- official, email, investigation

Example topics from real news with 100 number of topics:

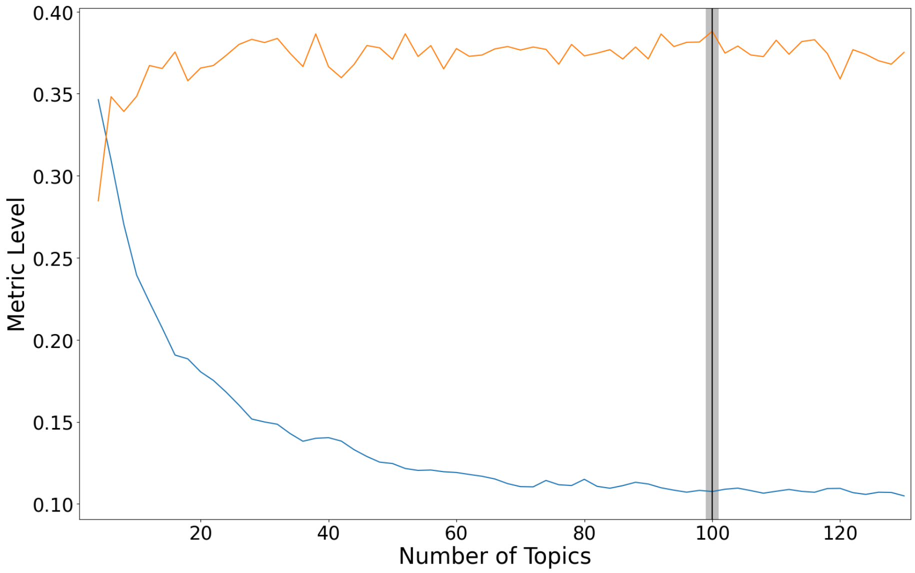
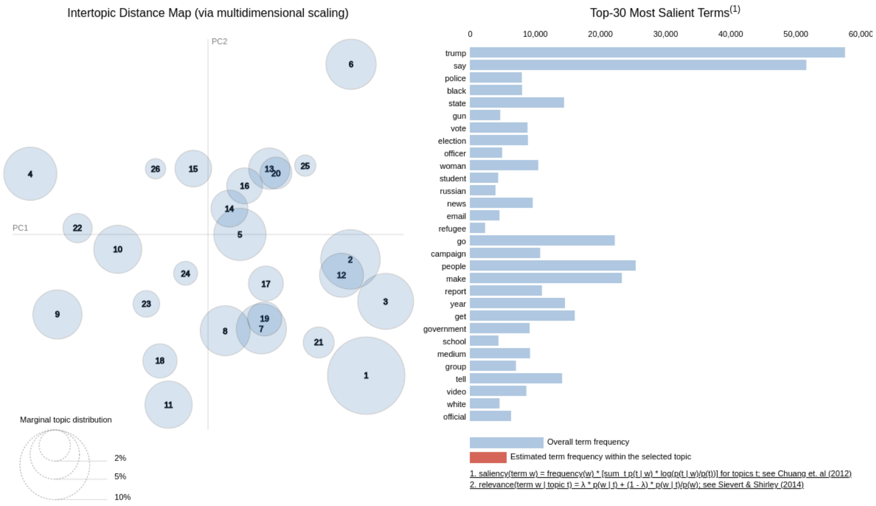


Figure 7: *Real news metrics*

- say, trump, candidate, presidential, campaign
- say, deal, agreement, trade, talk
- budget, say, year, plan, bill
- say, kill, attack, military, force
- say, trump, military, north_korea

Following are visualizations of topics with `pyLDAvis`[1] which is a python port of original LDAvis as published by Sievert and Shirley (2015). To interpret these visualizations, please see the LDAvis



1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

Figure 8: *Fake news topics*

[1] `https://github.com/bmabey/pyLDAvis`

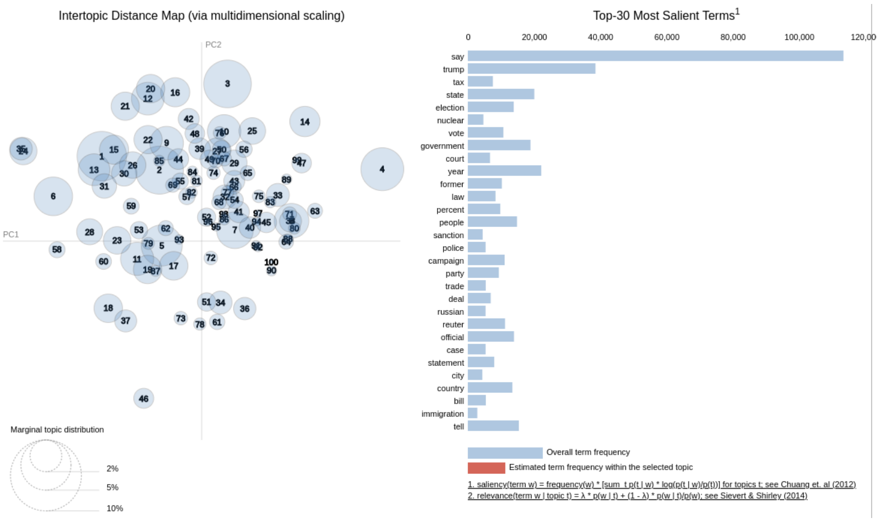


Figure 9: *Real news topics*

paper.

V. Discussion

In this paper, we explored the topics covered by fake news and compared with real news using topic modeling. We have found that fake news' topics are limited to some specific things, since the number of topics of fake news are less than that of real news. However, we found that it is hard to identify the topic from the generated set of words. And it is also possible that some sets of words are just noise or with no clear association with one another. The topic model just informs us expected topics, and it is up to us to judge whether those words really represent the meaningful topic. However, we are not experts of USA politics, so we could not clearly extract the genuine topics.

Extra work will be needed to improve LDA modeling accuracy. First, we need more data cleaning. Some words like 'say' can be generally used for almost every topic, so we should delete it from the bag of words. Second, the LDA model should be well-trained for better accuracy. To achieve it, we need to increase the training epochs. It is really time-consuming for it, but we could obtain a nice result. Next, we can try to use models other than LDA, such as Hierarchical Dirichlet Process Teh et al. (2006), Embedded Topic Model Dieng et al. (2020), Contextualized Topic Model Bianchi et al. (2021), or one of the newest BERTopic Grootendorst (2022). Lastly, we can ask an expert (of American politics) to evaluate our result and sort out unwanted noise topics.

The dataset we used might also be a limitation of our research. Although fake news are from unreliable sources, even collected from those, we cannot be sure that all the news labeled as 'fake' is really 'fake'. Thus, we should devise an alternative method to collect the fake/real news dataset.

The last limitation is that we cannot generalize our result to all fake news. What we have done is just analyzing fake news in U.S.(which half of the news had the word Trump). We cannot say that fake news in other countries also deals with fewer topics. Furthermore, there are various kinds of fake news besides political news, like entertainment, information news. So, it might be interesting to conduct our research for other kinds of fake news.

References

Ahmed, H., I. Traore, and S. Saad (2017). Detection of online fake news using n-gram analysis and machine learning techniques. Volume 10618 LNCS.

Ahmed, H., I. Traore, and S. Saad (2018). Detecting opinion spams and fake news using text classification. *Security and Privacy 1*.

Aletras, N. and M. Stevenson (2013). Evaluating topic coherence using distributional semantics.

Bianchi, F., S. Terragni, D. Hovy, D. Nozza, and E. Fersini (2021). Cross-lingual contextualized topic models with zero-shot learning.

Blei, D. M., A. Y. Ng, and M. I. Jordan (2003). Latent dirichlet allocation. *Journal of Machine Learning Research 3*.

Dieng, A. B., F. J. Ruiz, and D. M. Blei (2020). Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics 8*.

Greene, D., D. O'Callaghan, and P. Cunningham (2014). How many topics? stability analysis for topic models. Volume 8724 LNAI.

Grootendorst, M. (2022). Bertopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794*.

Honnibal, M., I. Montani, S. Van Landeghem, and A. Boyd (2020). spaCy: Industrial-strength Natural Language Processing in Python.

Jaccard, P. (1912). The distribution of the flora in the alpine zone.1. *New Phytologist 11*(2), 37–50.

Rehurek, R. and P. Sojka (2011). Gensim–python framework for vector space modelling. *NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic 3*.

Röder, M. (2015). Palmetto online demo. `https://palmetto.demos.dice-research.org/`.

Röder, M., A. Both, and A. Hinneburg (2015). Exploring the space of topic coherence measures.

Sievert, C. and K. Shirley (2015). Ldavis: A method for visualizing and interpreting topics.

Su, J., D. Greene, and O. Boydell (2016, December). Topic stability over noisy sources. In *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)*, Osaka, Japan, pp. 85–93. The COLING 2016 Organizing Committee.

Teh, Y. W., M. I. Jordan, M. J. Beal, and D. M. Blei (2006). Hierarchical dirichlet processes. *Journal of the American Statistical Association 101*.