# Counters: Identifying and Summarizing Opposing Media Articles

**Himank Yadav**
Cornell University
hy539@cornell.edu

**Katherine Van Koevering**
Cornell University
kav64@cornell.edu

## Abstract

'Fake news' is an increasingly visible problem, exemplified by articles with poor argumentation, suspicious sources, and unreliable evidence. However, many reliable news sources publish well-sourced articles on similar topics, either refuting or confirming points made by less reliable publishers. We attempt to find documents from reliable sources that are similar to a given unreliable article and then summarize these reliable documents. Finally, we use the summary as a counterpoint to the unreliable article.

## 1 Introduction

So-called 'fake news' has been an increasingly present topic in NLP research. The sheer volume of news articles available on the internet, means evaluating, detecting, and rebutting fake news have all been subject to attempts at automation, with varying results (Conroy et al., 2015).

However, there has also been work in examining news sources. While some sources are generally termed reliable (such as the New York Times), others are considered generally unreliable - whether because of bias, conspiracy theories, or other reasons. These unreliable news sources are widely available, and there has been concern about echo-chambers (Starbird et al., 2018), where a reader relies on less-reliable sources for news and may see the same falsehoods or inaccuracies over and over again. One possible way to ameliorate this effect is to automatically provide a summary of topics addressed in an unreliable article from a more reliable news source. This summary can then be used to rebut (or confirm) what is read in the unreliable news source. We hypothesize that it would be possible, through topic modeling, to identify reliable news articles on the same topics discussed in unreliable articles. We could then summarize these reliable articles to provide a counterbalance.

### 1.1 Inspiration and Related Work

This project is inspired by Neural Argument Generation Augmented with Externally Retrieved Evidence (Hua and Wang, 2018) by Xinyu Hua and Lu Wang. Using neural nets, they attempt to generate abstractive counterarguments to posts from subreddit r/cmv, an internet chat forum dedicated to debate on a variety of topics. This was done by first finding an opinion posted in the forum. From the topic signatures of this post, a set of queries would be constructed. These queries would then be used to retrieve relevant articles from Wikipedia, and then find relevant sentences within these articles. These sentences are then additional input to an abstractive generator to generate a counterargument to the post.

While we appreciate their technique for discovering relevant comments and information, we believe there is room for improvement in two particular areas: discovering the distance between a counterargument and a relevant argument, and the coherence of resulting arguments. This project is an attempt to address these concerns.

There has been previous work on automated fact checking. Hassan et al. used the 2016 US presidential debates as a data set to explore automatic fact checking of claims made by participants (Hassan et al., 2017). Karadzhov et al. looked at fact checking claims made by internet users, rather than articles, and also used reliable sources to counter-act these claims, but did so using a deep neural network (Karadzhov et al., 2017). In contrast, our work aims to primarily investigate media articles and reduce complexity by using simpler techniques such as topic modeling.

### 1.2 Unreliable News

In this project we avoid using the term 'fake news'. We intentionally refrain from passing judgment on the accuracy or quality of any individual article we use in our analysis. Instead, we recognize that various news sources are more reliable than others. While every news source makes mistakes, we expect that the amount of inaccuracies, poorly supported arguments, and bias is less for some news sources than for others. Thus we term some news sources 'reliable' and others 'unreliable'.

## 2 Data

While there are a number of datasts containing a variety of unreliable articles, we chose to use FakeNewsCorpus (Szpakowski, 2019), created by Maciej Szpakowski at University of Southampton under the supervision of Jonathon Hare. This corpus is a series of articles scraped from domains provided by OpenSources (curated by Melissa Zimdars from Merrimack College) that often publish unreliable news. It is supplemented by articles from the New York Times and Webhose English News Articles. The domains these articles are scraped from are broken down into categories: Fake News, Satire, Extreme Bias, Conspiracy Theory, State News, Junk Science, Hate News, Clickbait, Proceed With Caution, Political, and Credible. Discarding satire, junk science, and state news, we group the remaining types into two general categories: reliable and unreliable. In our case, Political and Credible are from 'reliable' news sources and the rest are not. All told there are approximately eight million articles in the corpus.

### 2.1 Sampling

Since we did not have the computational power necessary to include all 8 million articles in our dataset, we worked with a smaller sample. We randomly sampled each article in the data with a set probability of 0.05. This resulted in somewhere between 3,000 and 12,000 articles per type. We then did additional sampling to bring our totals to 10,000 articles per type. We also restricted our sample to only articles with at least 1000 characters, as we were primarily interested in longer articles.

### 2.2 Analysis

The length of the articles is heavily biased towards smaller articles. As stated before, we had a minimum length of 1000 characters, and the longest article in our dataset is 17,846 words. There does not seem to be too much of an obvious difference in the length of articles of different types, with perhaps the most striking difference being that 'reliable' articles are generally the shortest and have the smallest standard deviation. In contrast 'hate' articles have a very large standard deviation.

| Type | Mean | Standard Deviation | Median |
|---|---|---|---|
| bias | 790.4 | 915 | 524 |
| conspiracy | 877.7 | 971.6 | 627 |
| fake | 666 | 676.6 | 482 |
| hate | 805.6 | 1555.7 | 482 |
| unreliable | 876.3 | 824.2 | 677 |
| political | 753.1 | 781 | 524 |
| reliable | 619.6 | 601 | 463 |

Table 1: Token Length of Different Types

There is a much bigger difference in the domains that these articles come from. Articles of type 'fake', 'hate', 'unreliable', and 'political' all have one domain that dominates (beforeitsnews.com, amren.com, wikileaks.org, and dailykos.com, respectively). On the other hand, 'reliable', 'conspiracy', and 'bias' have more of a mix of domains where the articles are sourced from. It is unclear how this may have affected our analysis, and we would be curious in investigating further, as sites may publish frequently about the same topics or have similar writing styles. Fortunately, at least dailykos.com and beforeitsnews.com are known to be aggregators, frequently republishing articles from other sources. This makes it likely that their over-representation may not have a major effect. We suspect that wikileaks.org and amren.com may also be aggregators. Charts showing the proportion of articles sourced from different domains can be found in the appendix.

## 3 Methods

There are three steps in the methods we chose. First, we created a topic model using every article in our sample. Then, we chose a sample of articles from our 'unreliable' category. For each article, we selected a document or a few documents from our 'reliable' category that seemed to be about similar topics. We then created our summary from these documents.

### 3.1 Topics

We used MALLET to create a large LDA topic model of all of our documents. This included various preprocessing steps. First, we converted a given document into a list of tokens, where each token represented a word in the document. Accentuation was removed from all tokens. Next, we excluded all tokens with a length of than less 4 characters. Additionally, each token identified as a stop word from the stopwords list in the NLTK English Python package was excluded by the filtering process. Eventually, each token in our filtered list of tokens was lemmatized based on WordNetLemmatizer from the NLTK package as well. For each document, we created a dictionary object that encapsulated the mapping between our normalized tokens with their respective integer ids. Next, we converted each document into a bag-of-words format, and that represented our corpus.

We used MALLET (McCallum, 2002) for topic modeling. MALLET is a Java-based package that contains efficient, sampling-based implementations of Latent Dirichlet Allocation. We used a LDA model estimate that uses an optimized version of collapsed gibbs sampling from MALLET.

Our training corpus was rather large where the total number of tokens we trained on was 24,576,963 and the maximum number of tokens in the largest sentence we trained on was 9,293.

Each document consisted of a hundred topics, which was dependent to some extent on the size of our collection. We wanted to achieve fine-grained results while also ensuring that we had enough computing power to process the entire corpus. The number of sampling iterations for our topic model was 5000. Again, this was based on the time taken to complete sampling while also ensuring the quality of our topic model. We used hyperparameter optimization to allow the model to better fit the data by focusing on some topics over others. Such optimization was done every 10 iterations.

We initialized the hyperparameter alpha, which impacts the sparsity of the topics, to 0.005. The log likelihood of the model denotes how likely the data is given the model. Therefore, increasing log likelihood represents an improvement in the model. As shown in the graph in figure 1, our average log likelihood per token, i.e , the model's log likelihood divided by the total number of tokens, converged at -9.06471.

Additionally, we also tuned the hyperparameter

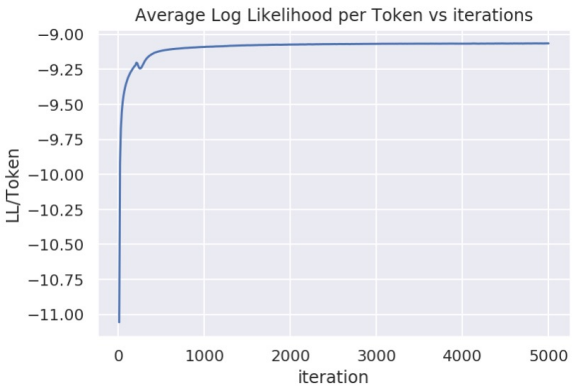



Figure 1: Average Log Likelihood

beta, which affects the collection of topics in each document. The lower the value of beta, the more concentrated the distribution of topics were in the model.

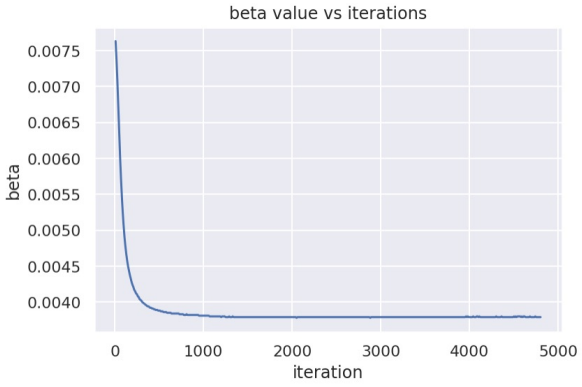



Figure 2: Beta Values

As shown in the graph in figure 2, beta converged at 0.00379, which is smaller than the MALLET default value of beta at 0.01.

### 3.2 Method 1: Topic-by-Topic

From previous experience with projects investigating misinformation and disinformation, we knew that articles from less reliable sources could frequently cover multiple, seemingly unrelated, topics and ideas. Since it seemed unlikely that any single reliable article would address these, we treated each topic in our unreliable article separately.

First, we created a topic-doc matrix that gave the three reliable documents with the highest score for each topic. We then took each unreliable article, retrieved its topic vector, and recorded the topics for which the document scored above .15.

For each of those topics, we retrieved the reliable documents from the topic-doc matrix. If the first document had a length of less than 500 words, we appended the second document. If the combination of those two documents still had less than 500 words, we appended the third document. This would be the summarizing text for that topic. We then concatenated all the summarizing text for every topic above the threshold in the document. This text was then passed to the summarizer.

### 3.3 Method 2: KDTree

The second method also utilized the topics we determined using MALLET. In this case, rather than using articles that exemplify each topic, we try to match documents based on the topic vector as a whole. We create a KDTree using the topic vectors of all 20,000 trusted documents. Once we have this tree, we can then query it with topic vectors of untrusted documents for the most similar articles in the tree. We take the 5 most similar trusted documents from the tree and, employing the same method above, concatenate them until we hit at least 500 words. This concatenated text is then passed to the summarizer.

### 3.4 Method 3: TF-IDF

This last method we employ actually does not look at topics at all. Instead, we compare the TF-IDF vector as a measure of document similarity. We fit on all 70,000 documents and then transform the 20,000 trusted documents and use their TF-IDF vectors to train a K-NN classifier, where each point is in its own class. By then feeding the K-NN classifier the TF-IDF vectors for untrusted documents and asking the K-NN to classify the vector using only the 1st nearest neighbor, the model gives us the closest reliable document. We then take just this document as our text to summarize, regardless of the length of the text.

### 3.5 Summarization

Since one of the main goals of this project was to improve coherence as compared to Hua and Wang's counterargument generation, we decided that extractive summarization would give us more reliably coherent summaries. First, we concatenate all of the documents retrieved for the summary. We use TextRank to summarize the articles, with a target length of 10% the full length of the retrieved articles (Mihalcea and Tarau, 2004). Since TextRank does not depend on order, we simply concatenate the articles in the order they were retrieved. Most summaries have a length of 3-10 sentences, with rare summaries being much longer. We decided against having a set target length, since an untrustworthy article with more topics would retrieve more articles, and so should have a longer summary.

## 4 Evaluation

We evaluated using a small sample of summarized work. For each type of untrustworthy article, we randomly sampled 30 articles to create counterpoint summaries for. These 30 articles are sampled from the 10,000 articles included in our topic modeling for that type.

However, although the documents in our dataset appeared fairly clean, there were a number that when examined more closely were unsuitable. Usually, the reason for this was the format of the document. For instance, a legislative bill or a comments section. However, there were also a few documents that were not in English. Since we did not catch these in our first run through the data, we threw them out during the evaluation stage, and replaced them with additional random samples from the same category. In total, we replaced 28 articles.

### 4.1 Human Evaluation

There are no gold standard summaries for these articles, and we recognize the difficulty of machine evaluation of summarization and text generation. Thus, human evaluation seemed the best way to get an accurate representation of how well the techniques worked.

We evaluated our summaries based on four categories: coherence, coverage, extraneous information, and repetition, scoring them on a scale from 1-5. Extraneous information was defined as information unrelated or only tangentially related to any of the points in the article. A coherence score of 1 meant it was unintelligible, a coherence score of 5 meant the summary felt like it was written by a person. A coverage score of 1 meant the summary did not cover any of the main points of the article. A coverage score of 5 meant the summary covered all the main points of the article. An extraneous information score of 1 meant the summary consisted almost entirely of extraneous information. An extraneous information score

of 5 meant the summary contained no extraneous information. A repetition score of 1 meant the summary was merely the same sentence repeated many times. A repetition score of 5 meant the summary did not repeat any information. We first read the article, taking note of the main points it brought up, then read the summary. We then immediately scored the summary. We also noted down the 'theme' of the article, and the 'type' of the article (opinion, event, interview, etc.).

### 4.2 Machine Evaluation

We used standard measures for measuring coverage, extraneous information, and repetition in our summaries. BLEU (Papineni et al., 2002) scores are frequently used as an evaluation metric in Machine Translation tasks but they are also helpful while evaluating machine summarization since they perform $n$-gram comparison between words in candidate and reference sentences. In our case, BLEU score reflects the $n$-gram overlap between our generated summary and the original document. BLEU-1 refers to a 1-gram or a unigram overlap between generated summaries and the original document while BLEU-2 refers to a 2-gram or a bigram overlap between generated summaries and the original document. BLEU-3 and BLEU-4 scores are calculated similarly. As seen in Table 2, our performance decreases as $n$ in BLEU-$n$ increases since the complexity of the metric increases drastically. Both BLEU and GLEU score ranges are always between 0 (no matches) and 1 (all match).

Next, we evaluate our output using GLEU (Google-BLEU) (Wu et al., 2016) scores. In our case, GLEU computes all sub-sequences of lengths varying from 1 to 4 in both generated summaries as well as the original document and computes a recall and precision for these $n$-grams. The GLEU score is then calculated by taking the minimum of recall and precision. As the authors state, GLEU scores overcome some shortcomings of BLEU since they correlate well with BLEU at a corpus level but do not contain some of the drawbacks of BLEU at a sentence level.

Since our summaries are generated from multiple trustworthy documents from various sources, we wanted to evaluate our summaries using METEOR (Banerjee and Lavie, 2005) scores as they claim to be closer to human judgement. METEOR overcomes a shortcoming of BLEU while dealing with individual sentences and also takes into account word stemming and synonyms.

Lastly, we evaluate using ROUGE (Lin, 2004) scores. ROUGE is mainly based on recall and we consider three different types of ROUGE scores. ROUGE-N is computed by recall based on matching $n$ grams. We evaluate our summaries based on unigrams (ROUGE-1) and bigrams (ROUGE-2). Finally, we also evaluate using ROUGE-L which uses the longest common sub-sequences and the F-scores based on precision and recall for ROUGE-L as shown in Table 2.

## 5 Experiments

Brief evaluation of the results of the topic-by-topic method and the KDTree method showed a clear improvement using the KDTree. Not only were the summaries more coherent, but they were generally about much more similar topics to the original article than the topic-by-topic method. It is possible that this is due to the breadth of our topics. Since our data set was so varied, our topics were quite broad, for instance we had topics around crime, Trump, and Israel. More specific topics might have been things like Hamas, Obama's birth certificate, or Black Lives Matter. These broad topics likely negatively impacted our topic-by-topic summaries as they were less coherent and very general. It seemed to have less of an impact on our KDTree method, although it is not clear how the KDTree method might have performed with more specific topics. Since it was so clear that the KDTree method performed better, we only formally evaluated that method 3.

### 5.1 KDTree

While the summaries were often about the same general topic or a very similar topic to the original article, it was rare that the summary really addressed the same points as the original article. Cases where this did happen seemed to be largely due to chance, rather than a well performing system. Additionally, there did not seem to be any significant difference in performance across the types of articles (bias, conspiracy, etc.).

For the most part, the summarization method seemed to work very well. The summaries scored very highly in coherence and repetition across the types. Although we did not specifically code for this, it was also apparent that the summarization effectively captured most of the main points of

| | Mean | Median | Max | Min | Stdev |
|---|---|---|---|---|---|
| BLEU-1 | 0.146 | 0.021 | 0.862 | 0.0 | 0.232 |
| BLEU-2 | 0.115 | 0.016 | 0.661 | 0.0 | 0.179 |
| BLEU-3 | 0.067 | 0.011 | 0.423 | 0.0 | 0.103 |
| BLEU-4 | 0.036 | 0.004 | 0.241 | 0.0 | 0.056 |
| GLEU | 0.079 | 0.012 | 0.486 | 0.0 | 0.124 |
| METEOR | 0.065 | 0.051 | 0.2 | 0.0 | 0.041 |
| ROUGE-1 | 0.11 | 0.112 | 0.278 | 0.0 | 0.056 |
| ROUGE-2 | 0.013 | 0.012 | 0.086 | 0.0 | 0.012 |
| ROUGE-L | 0.072 | 0.066 | 0.216 | 0.0 | 0.046 |

Table 2: Machine evaluation results for KDTree method

| Type | Measure | Coherence | Coverage | Repetition | Extraneous.Information |
|---|---|---|---|---|---|
| Fake News | | | | | |
| | Minimum | 2 | 1 | 1 | 1 |
| | Median | 4 | 3 | 5 | 2 |
| | Mean | 4.103448 | 2.551724 | 4.689655 | 2.551724 |
| | Maximum | 5 | 5 | 5 | 5 |
| | Stdev | 1.080503 | 0.909718 | 0.806379 | 1.212618 |
| Extreme Bias | | | | | |
| | Minimum | 2 | 1 | 4 | 1 |
| | Median | 4 | 2 | 5 | 2 |
| | Mean | 3.933333 | 2.2 | 4.833333 | 1.9 |
| | Maximum | 5 | 4 | 5 | 4 |
| | Stdev | 1.112107 | 0.961321 | 0.379049 | 0.959526 |
| Conspiracy Theory | | | | | |
| | Minimum | 2 | 1 | 1 | 1 |
| | Median | 4 | 2 | 5 | 2 |
| | Mean | 4.033333 | 2.166667 | 4.3 | 1.9 |
| | Maximum | 5 | 4 | 5 | 4 |
| | Stdev | 1.098065 | 0.949894 | 1.207734 | 0.959526 |
| Hate News | | | | | |
| | Minimum | 2 | 1 | 2 | 1 |
| | Median | 4 | 2 | 5 | 2 |
| | Mean | 3.8 | 2.2 | 4.6 | 2 |
| | Maximum | 5 | 4 | 5 | 4 |
| | Stdev | 0.846901 | 1.030567 | 0.723974 | 0.946864 |
| Unreliable | | | | | |
| | Minimum | 2 | 1 | 4 | 1 |
| | Median | 4 | 2 | 5 | 1 |
| | Mean | 4 | 2.103448 | 4.862069 | 1.862069 |
| | Maximum | 5 | 4 | 5 | 5 |
| | Stdev | 0.92582 | 0.976321 | 0.350931 | 1.27403 |
| Full Corpus | | | | | |
| | Minimum | 2 | 1 | 1 | 1 |
| | Median | 4 | 2 | 5 | 2 |
| | Mean | 3.972973 | 2.243243 | 4.655405 | 2.040541 |
| | Maximum | 5 | 5 | 5 | 5 |
| | Stdev | 1.009788 | 0.966244 | 0.77996 | 1.093446 |

Table 3: Human evaluation results for KDTree method

| | |
|---|---|
| Opinion | 42 |
| Event | 60 |
| Investigation | 17 |
| Interview | 4 |
| Report | 12 |
| Total | 135 |

Table 4: Most common categories of articles

the text summarized. This is especially impressive since many of our summarizing texts were actually concatenations of multiple, sometimes very different, articles. We would recommend using TextRank as a summarization tool in any project in a similar vein.

However, the summaries scored very poorly in coverage and extraneous information. It was rare for a summary to score above a 2 in either category, and it seems likely that most high scores are due to chance rather than a functioning system. However, as stated before, the summaries were often about very similar topics if not actually addressing the particular points of the article, implying that there is at least some relation between the summaries and the articles.

### 5.2 Categories

While evaluating we additionally recorded the category in which we felt the article fit 4. An 'opinion' article primarily discusses the authors opinion or personal feelings, and is usually not about a specific event. We defined 'event' as an article about a particular, usually recent, event in time. For instance, an arrest, the release of a report, or a fire. An 'investigation' would be an in-depth investigation into a series of events, such as an article about GMO crops or the Syrian war. An 'interview' would primarily consist of either the raw text of an interview or commentary on an interview. A 'report' would be something like a financial summary, or official documentation. Most 'report' articles were thrown out as not within the scope of the project. In total, 135 of 150 articles that were kept fell into one of these categories.

While we had expected the vast majority of articles to fall under the 'event' category, this turned out not to be the case. There were almost as many 'opinion' articles as there were 'event' articles. Additionally, many 'event' articles were quite similar to the 'opinion' category. Even when an article primarily reported on an event, much of the article would be spent talking about the author's personal opinions, theories, or anecdotal stories. This was not expected and may have contributed to the poor results above.

### 5.3 TF-IDF

Additionally, we also briefly compared the KDTree method to the TF-IDF method. First, we randomly selected 20 articles that had been evaluated for the KDTree method. We then used the same human evaluation metrics previously described to evaluate the summaries produced 5. Again, there was not a significant difference in performance over the KDTree method. In fact, the metrics looked remarkably similar.

Additionally, the TF-IDF summaries tended to be more correct in details and less correct in subject. For example, an article about immigration may contain statements from Obama. While the KDTree method will likely produce a summary roughly around immigration, a TF-IDF summary may contain statements from Obama about something different, such as climate change. This may indicate that the TF-IDF and KDTree methods are capturing different aspects of the articles and combining them may lead to an overall improvement in summary quality.

## 6 Conclusion

Our results indicate that our methods independently were not very effective in our stated goal - a 'fact checking' device for untrustworthy articles. However, portions of these methods show promise. As stated above, while the summaries rarely succeeded in capturing the main points of the articles, they were frequently able to capture the theme of the article. This could imply that topic modeling or TF-IDF could be part of a future, more effective method.

Additionally, we have shown some of the weaknesses involved in attempting to identify reliable articles as a counterpoint to unreliable articles. Primarily, many unreliable articles are quite messy, possibly making it difficult to automatically decipher the main points. It is also clear that topics addressed are far more varied than we anticipated and it may be necessary to focus further efforts to one or two broad topics.

| | Coherence | Coverage | Repetition | Extraneous Information |
|---|---|---|---|---|
| Min | 2 | 1 | 2 | 1 |
| Median | 4 | 2 | 5 | 1 |
| Mean | 4.05555 | 2.111111 | 4.38888 | 1.94444 |
| Max | 5 | 4 | 5 | 4 |

Table 5: Human evaluation results for TF-IDF method

## 7 Future Work

We believe there may be promise in the methods we used. Perhaps the biggest improvement could be achieved by using a different, cleaner, data set. Most of the untrustworthy articles in our data we would not classify as a typical news article. A large number are opinion pieces, with more text devoted to the author's personal opinions or humor than the actual event or subject being discussed. This likely had an impact on how well we could determine the subject of the article, but also whether or not there even exist reliable articles that discuss the same ideas. However, it is also true that these are the kinds of articles that exist in the wild, and so to create a truly effective model, one would have to learn how to handle these sorts of texts.

It also seems that there are articles that are specific enough we do not have trustworthy articles about the same set of topics for other reasons. Although we initially hypothesized that it would be difficult to find exact matches for untrustworthy articles among trustworthy articles, we underestimated this effect. We expect merely narrowing our field of exploration (for instance, just focusing on political content) would have helped with this, both allowing for more precise topics and more reliable articles on particular precise subjects. However, it is also possible that a much larger set of trustworthy articles may be necessary, to the point where the search space becomes difficult to work with. It is also possible that the necessary articles just don't exist, especially as the definition of a reliable source narrows, and more compositional techniques may be necessary.

Another possibility is combining two of our methods. While neither the topic modeling nor the tf-idf vectorizer were perfect, they each seemed to capture different aspects of the articles. Topic modelling captured the overall theme of the piece (immigration, race relations, religion, environmentalism, etc.), while tf-idf captured more specifics (BP oil-spill, Raqqa, etc.). Somehow combining these could perhaps improve how we pinpoint articles that capture both the theme and the specifics.

## 8 Division of Labor

Himank: Sampling, Topic modeling, MALLET, Topic-by-Topic, KDTree, human evaluation, machine evaluation Katherine: Sampling, Topic-by-Topic, KDTree, TF-IDF, Summarization, Human Evaluation & Analysis, Data Analysis, Paper 1st draft

## 9 Acknowledgements

We would like to thank Xanda Schofield for her insight on topic modeling.

## References

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*. pages 65–72.

Niall J Conroy, Victoria L Rubin, and Yimin Chen. 2015. Automatic deception detection: Methods for finding fake news. *Proceedings of the Association for Information Science and Technology* 52(1):1–4.

Naeemul Hassan, Fatma Arslan, Chengkai Li, and Mark Tremayne. 2017. Toward automated fact-checking: Detecting check-worthy factual claims by claimbuster. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, pages 1803–1812.

Xinyu Hua and Lu Wang. 2018. Neural argument generation augmented with externally retrieved evidence. *arXiv preprint arXiv:1805.10254* .

Georgi Karadzhov, Preslav Nakov, Lluís Màrquez, Alberto Barrón-Cedeño, and Ivan Koychev. 2017. Fully automated fact checking using external sources. *arXiv preprint arXiv:1710.00341* .

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out* .

Andrew Kachites McCallum. 2002. Mallet: A machine learning for language toolkit .

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, pages 311–318.

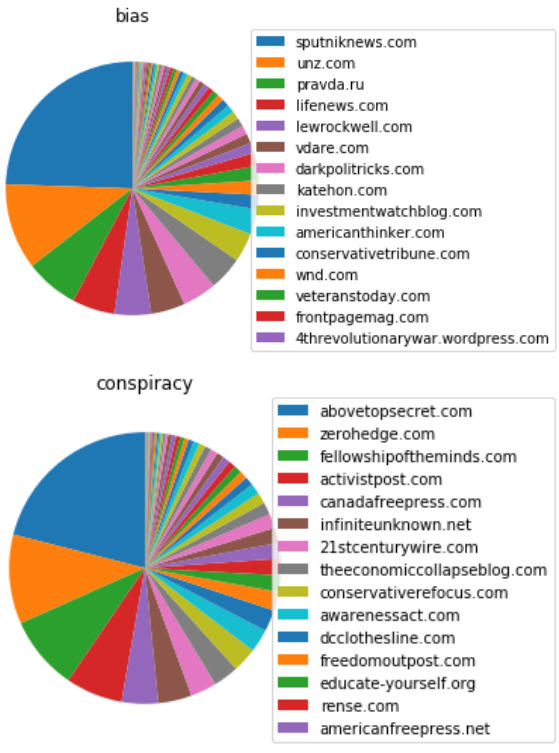
Kate Starbird, Ahmer Arif, Tom Wilson, Katherine Van Koevering, Katya Yefimova, and Daniel Scarnecchia. 2018. Ecosystem or echo-system? exploring content sharing across alternative media domains. In *Twelfth International AAAI Conference on Web and Social Media*.

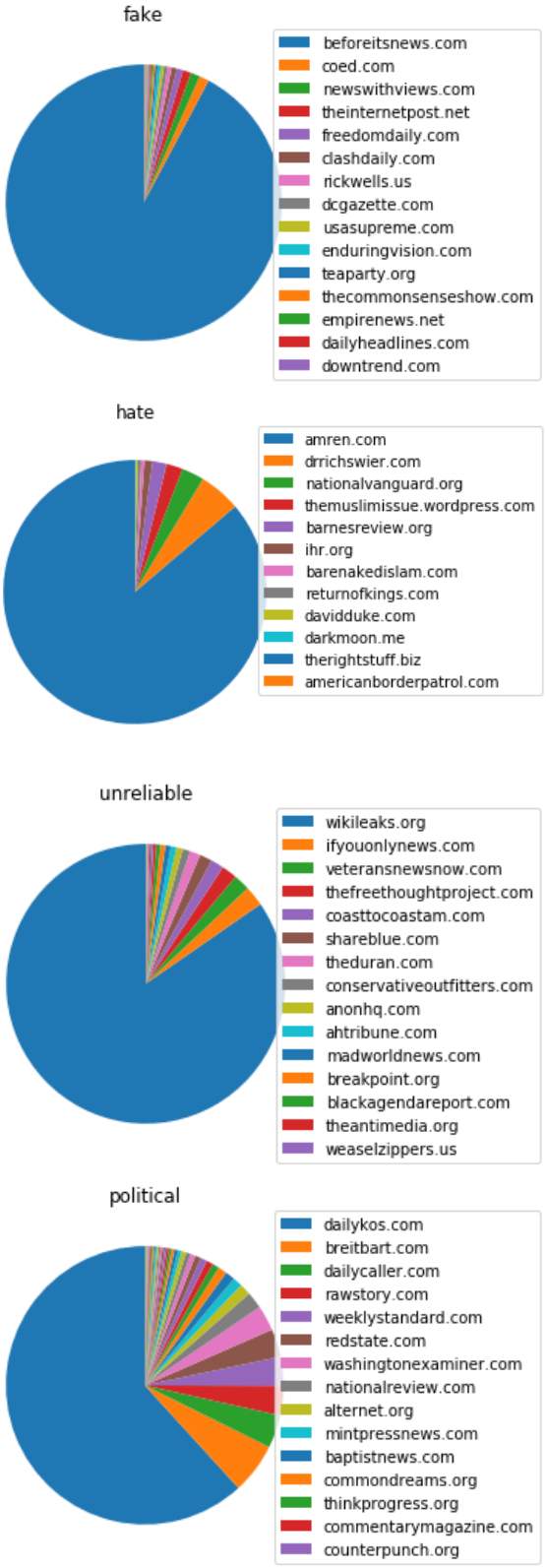
Maciej Szpakowski. 2019. Fakenewscorpus. https://github.com/several27/FakeNewsCorpus.

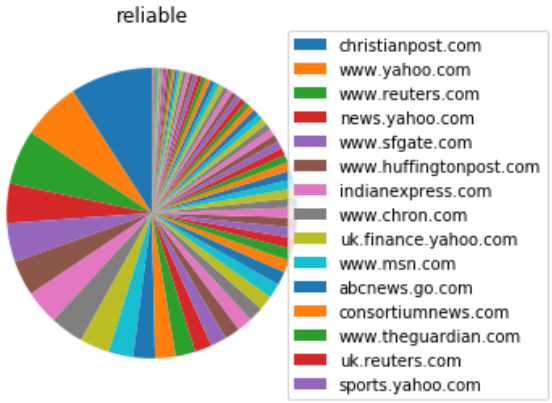
Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144* .

# A Appendix

## A.1 Article Domains




reliable
christianpost.com
www.yahoo.com
www.reuters.com
news.yahoo.com
www.sfgate.com
www.huffingtonpost.com
indianexpress.com
www.chron.com
uk.finance.yahoo.com
www.msn.com
abcnews.go.com
consortiumnews.com
www.theguardian.com
uk.reuters.com
sports.yahoo.com