Matthew Thomas (met8k@virginia.edu)
DS5001
Final Project Report
4/30/20

Abstract

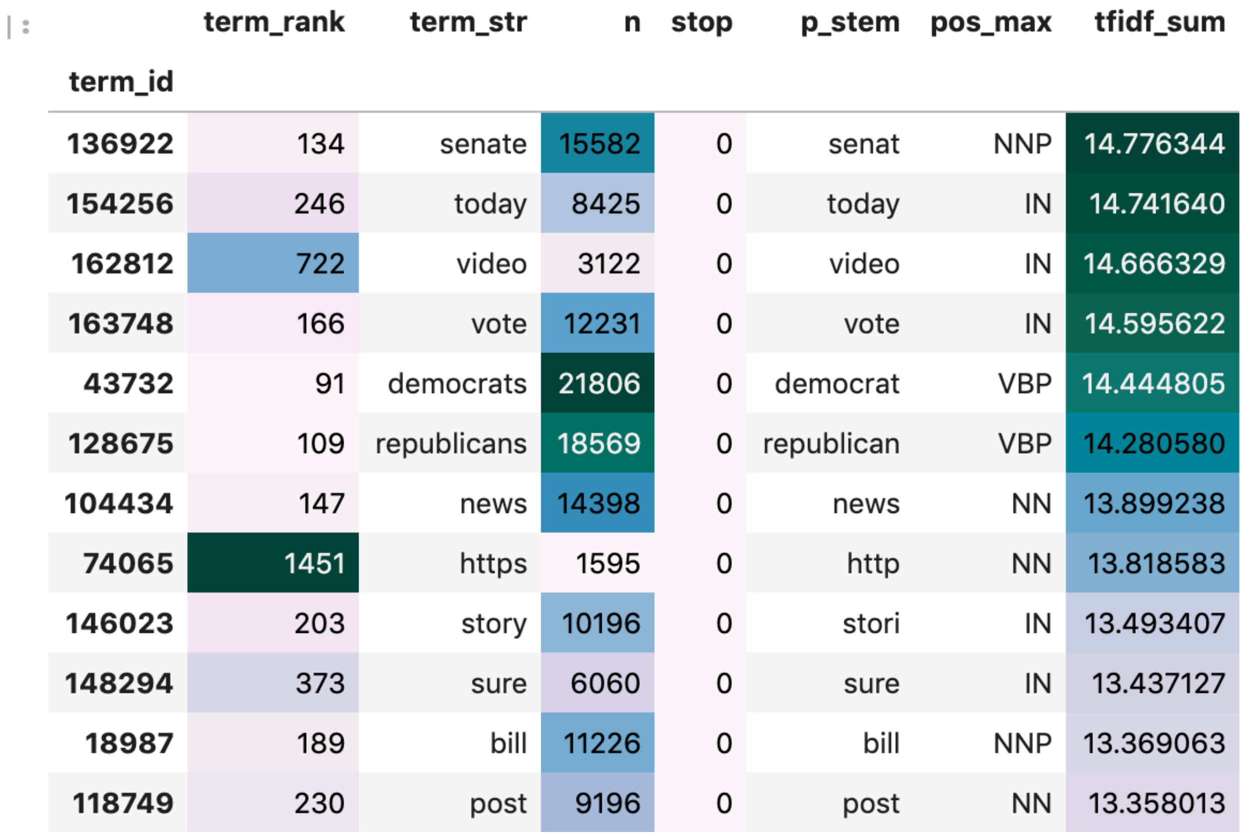
The goal of this project is to perform Exploratory Text Analytics on both conservative and progressive news sources found on the internet over a period of several years. The focus was on three sources: Powerline, a conservative news site, The Daily Kos, a progressive/liberal news site and blog, and Politico, a somewhat progressive news site (*note: ideally I would have liked a second conservative news site, but computer limitations regarding data size and problems scraping the Breitbart site prevented that).* I scraped articles from Politico and Daily Kos and collected them into a corpus along with Powerline articles already provided. The articles ranged in date from November 2013 to February 2020, although not all three publications covered that entire time span. I also scraped articles from Real Clear Politics, but many of the articles were aggregated from other news sites such as the New York Times and the Atlantic, so this site was not used. This was an unsupervised learning project with no particular hypothesis being tested. Rather, the goal is to use PCA, word embeddings, TF-IDF tables, and topic models to uncover insights into what issues these sites are concerned with and how they cover those issues. As we shall see, the word embeddings were useful in highlighting differences in word usage and associations, and the topic models in particular gave insight into what issues were covered the most, especially for the more partisan sites.

Pre-Processing

Pre-processing consisted of filtering out words that were not relevant to the task, such as html commands and author names (since not all publications included author names, there were no useful comparisons to be made). The articles were separated by sentence and token, so that the OHCO was Source-Article-Sentence-Token. Ideally paragraphs would have been included but the scraping tool returned articles as one large paragraph. The tokens were tagged with part-of-speech and then made lowercase with punctuation removed. Finally, very short articles were removed since these were either bad links or conveyed little information.

TF-IDF

To compute the TF-IDF table, I used a function and created three different TF-IDF tables using different metrics to compare. Because the tables were very large, it wasn't feasible to compute more than that. The first one was a raw count of term frequency and the standard IDF formula $\log\frac{N}{DF}$. The second was the raw count TF and the 'smooth' IDF $\log\frac{N}{(1+DF)}+1$ and the third was the sum term frequency and the probabilistic IDF $\log\frac{N-DF}{DF}$. The sum of the TF-IDF values were aggregated on the vocabulary table to see which words had the highest rating. The first two yielded similar results with the highest rated words being standard verbs. I picked the third table because the highest rated words were more relevant to the source, e.g. republican, senate, democratic, media. I then calculated an entropy for each term on the vocabulary table and used those values to filter out low-information words as well as frequently used words to bring the TF-IDF table to a manageable size. Below is an image of part of the VOCAB table with highest TF-IDF sum words at the top. Most word meanings are obvious. ‘Post’ probably refers to the Washington Post.



	term_rank	term_str	n	stop	p_stem	pos_max	tfidf_sum
term_id							
136922	134	senate	15582	0	senat	NNP	14.776344
154256	246	today	8425	0	today	IN	14.741640
162812	722	video	3122	0	video	IN	14.666329
163748	166	vote	12231	0	vote	IN	14.595622
43732	91	democrats	21806	0	democrat	VBP	14.444805
128675	109	republicans	18569	0	republican	VBP	14.280580
104434	147	news	14398	0	news	NN	13.899238
74065	1451	https	1595	0	http	NN	13.818583
146023	203	story	10196	0	stori	IN	13.493407
148294	373	sure	6060	0	sure	IN	13.437127
18987	189	bill	11226	0	bill	NNP	13.369063
118749	230	post	9196	0	post	NN	13.358013

PCA

I performed PCA with 10 components on the TF-IDF table using the Scikit-Learn library after normalizing for vector length. Examining the words on the extreme ends of the loadings, it’s difficult to find an interpretation but worth noting that for both the first and second PC there is an emphasis on the Russia election investigation. Visualizing the PCs made things slightly clearer. Looking at the first PC, stories on the negative end are mainly about Russia and impeachment, while stories on the positive end are about elections and the political parties. The second PC is less clear but many of the articles on the negative end are about health care. There was a lot of overlap between the second and third PCs and interpretation is unclear.

One interesting find is the overlap between news sources for the first PC. There is a clear separation between Powerline and Politico, while Daily Kos is more evenly spread. A possible reason is that Powerline and Daily Kos, being more highly partisan, were more concerned about impeachment and the Russia investigation, while Politico was more concerned about elections.

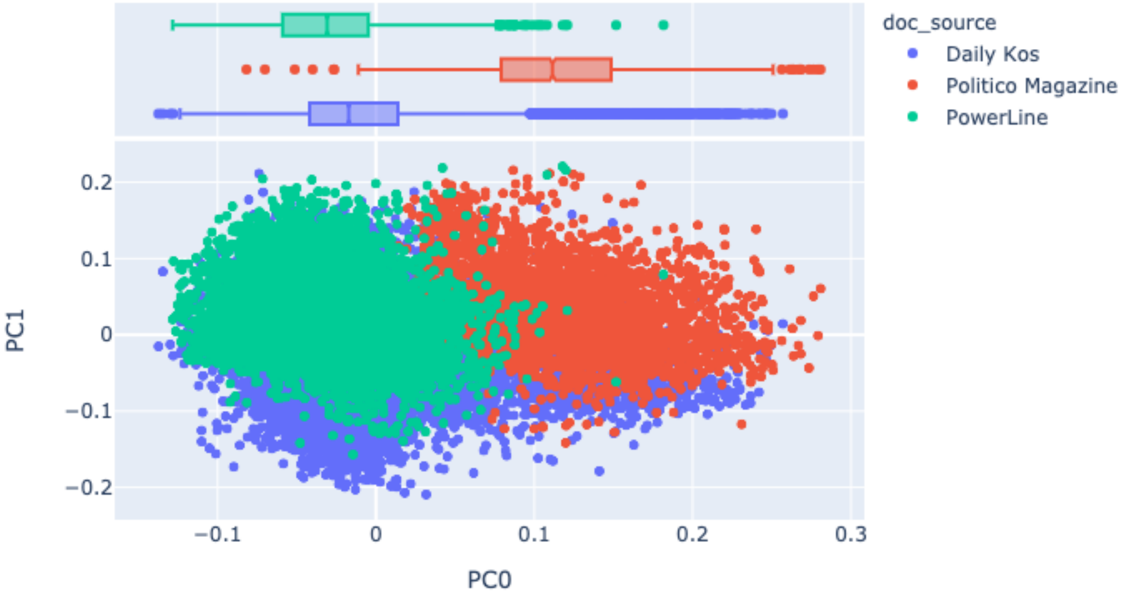


Figure 1 PCA Scatterplot

Word Embeddings

To find word embedding vectors, I ran word2vec first on the corpus as a whole and then on each separate news source. For each model, I made a TSNE visualization of the words to see how the model grouped them. Running the model on the full corpus didn't reveal much, the model simply grouped terms by similar topics (it is also difficult to visualize because of the volume of words). Visualizing the models of each news source was more interesting because the word groupings revealed contexts regarding contentious keywords. For example, Powerline groups “Hillary” and “Clintons” close to words such as “opposition”, “foreign”, “Russian”, “Ukraine”, and “corruption”, whereas Politico grouped those words close to other candidates (“Sanders”, “McCain”) and words about elections. Clearly Powerline mentioned Hillary Clinton much more in the context of the Russia investigation, which of course was more about Trump (although Trump does not appear near these words).

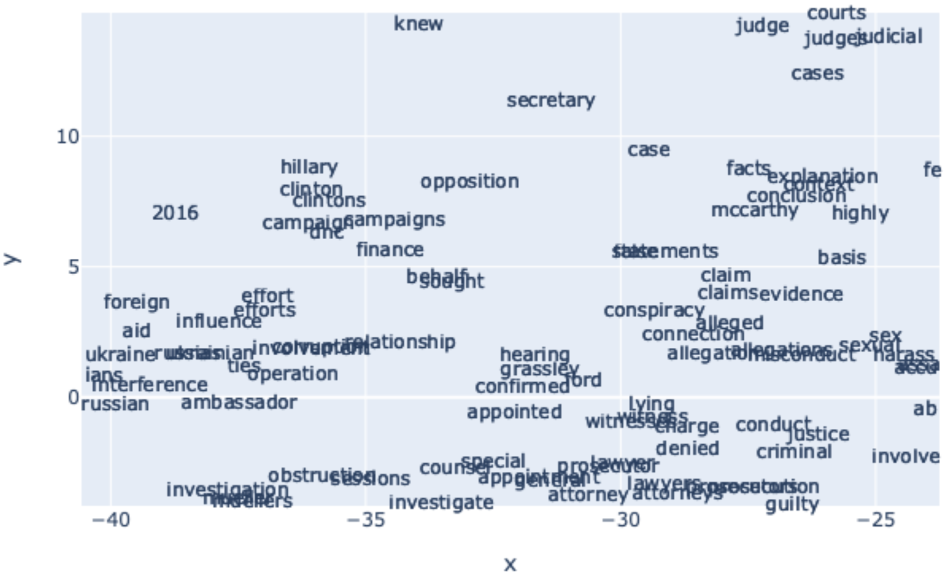


Figure 2 TSNE plot section Powerline

Examining similar words based on vector space distance and analogies using semantic algebra was also revealing. Finding words similar to “abortion”, both Daily Kos and Politico grouped in gender or family words (“marriage”, “parenthood”, “transgender”), medical words, and “marijuana” for unclear reasons. Powerline’s words were related to law and democracy (“majority”, “minority”, “laws”, “opposed”, “vast”). Another good comparison was the most similar words to “racism”. Powerline included words such as “white”, “hatred”, “color”, “culture”, and “violence”, whereas Kos and Politico had words such as “identity”, “slavery”, and “supremecy”, indicating that these websites view the issue very differently (cultural versus historical). Politico’s words also included “jews” and “muslims”, which shows a focus on racism towards or among these groups in particular. Most analogies weren’t illuminating, but an interesting analogy was “republican:conservative, democrat: “. All included “liberal” or “liberals” but Powerline also had “leftists” whereas Daily Kos had “moderate”, indicating that Powerline regards Democrats as being radical and Kos regards them as in the center. See notebook for additional examples.

Sentiment Analysis

Sentiment Analysis on news sources is difficult because even opinion columns don’t use many emotion words. To gauge sentiment, I compared the three news sources using the NRC lexicon and then with the vader python package. The results were very similar using the NRC lexicon across all three sources, but it is worth noting that Politico had lower levels of anger and sadness compared to the others, which were nearly exactly the same.

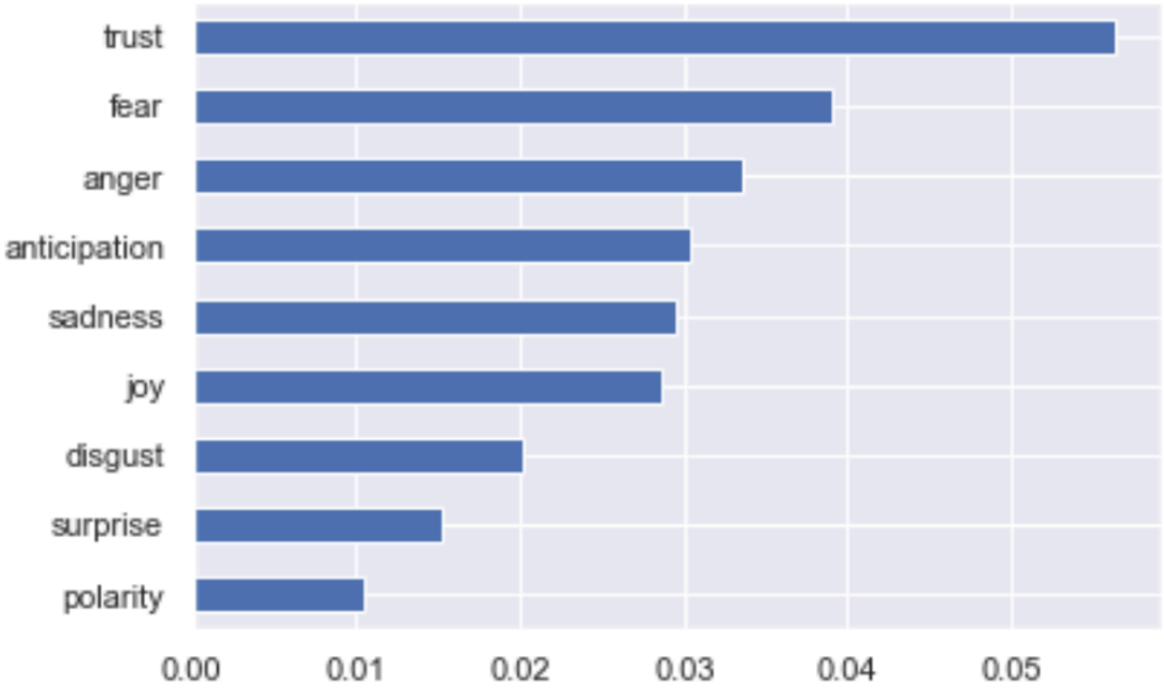


Figure 3 Politico Sentiment NRC Lexicon

I used the vader package to plot sentiment by date of the article to judge at which times sentiment was highest or lowest, as in the examples below.

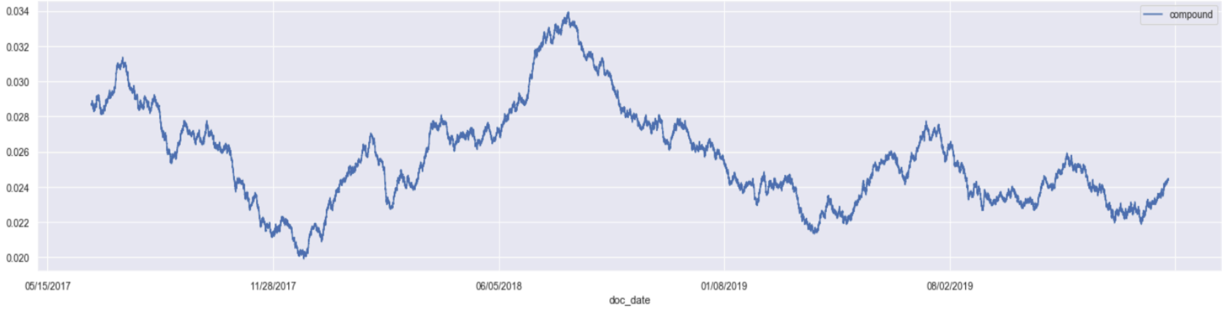


Figure 4 Powerline Sentiment Indexed by Date

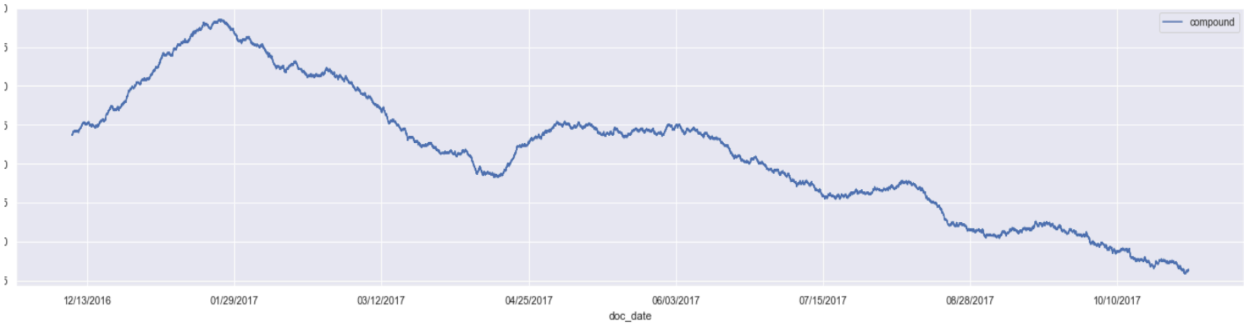


Figure 5 Daily Kos Sentiment Indexed by Date

Although we can clearly see changes in sentiment over time, interpreting the reasons is difficult. In the case of Daily Kos, perhaps it’s fatigue from talking about the Trump administration. Powerline is also unclear, except perhaps for a decline in sentiment coinciding with Democrats’ victory in the 2018 midterms as well as the Mueller investigation in 2019. Politico’s sentiment (see notebook) rose steadily over time but determining why would require closer examination of the articles.

Topic Models

To find topic models, I used the Mallet package and ran it on the reduced token table (aggregated to articles) with 50 topics (this number was arbitrary but given the size of the corpus it seemed appropriate). The Mallet package uses Gibbs sampling and Latent Dirichlet Allocation to find the topics and topic words. I analyzed the DOCTOPIC table, which shows the weights of each topic across all documents (articles), along with the TOPIC table that contains the top words for each topic as well as an inferred topic label. Finally, I used the TOPICWORD table, which when transposed becomes a PHI table, to cluster the topics.

Not all of the topics were interpretable, but most had an easy interpretation given what we know about current events. For example, topic 0 (see notebook) is clearly about the Trump impeachment proceedings, topic 23 is about big tech, topic 2 is about health care, and so on.

Grouping the topic weights by news source, nearly all of the highest weights for each topic were Daily Kos and Powerline, indicating that these sources probably discussed the same events day after day, while Politico had a broader range of coverage (see notebook for full chart). Clustering the topics using the PHI table and measuring by Euclidean distance, the program was able to cluster topics into coherent meta-topics. For example, the image below is one of the two big clusters. The green cluster is about Trump and the Mueller/Russia investigation, the black cluster is foreign policy, and the yellow cluster can be called elections. It’s particularly impressive that the computer was able to lump together topic 36, which is North Korea and China, with the topics about the Middle East.

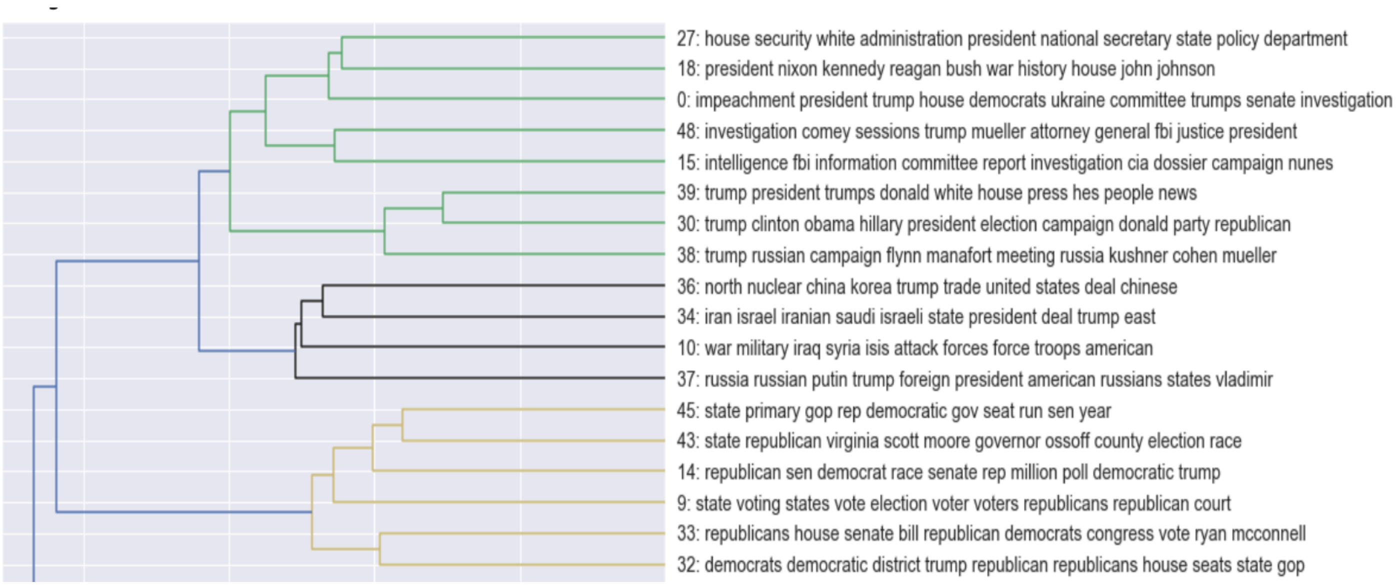
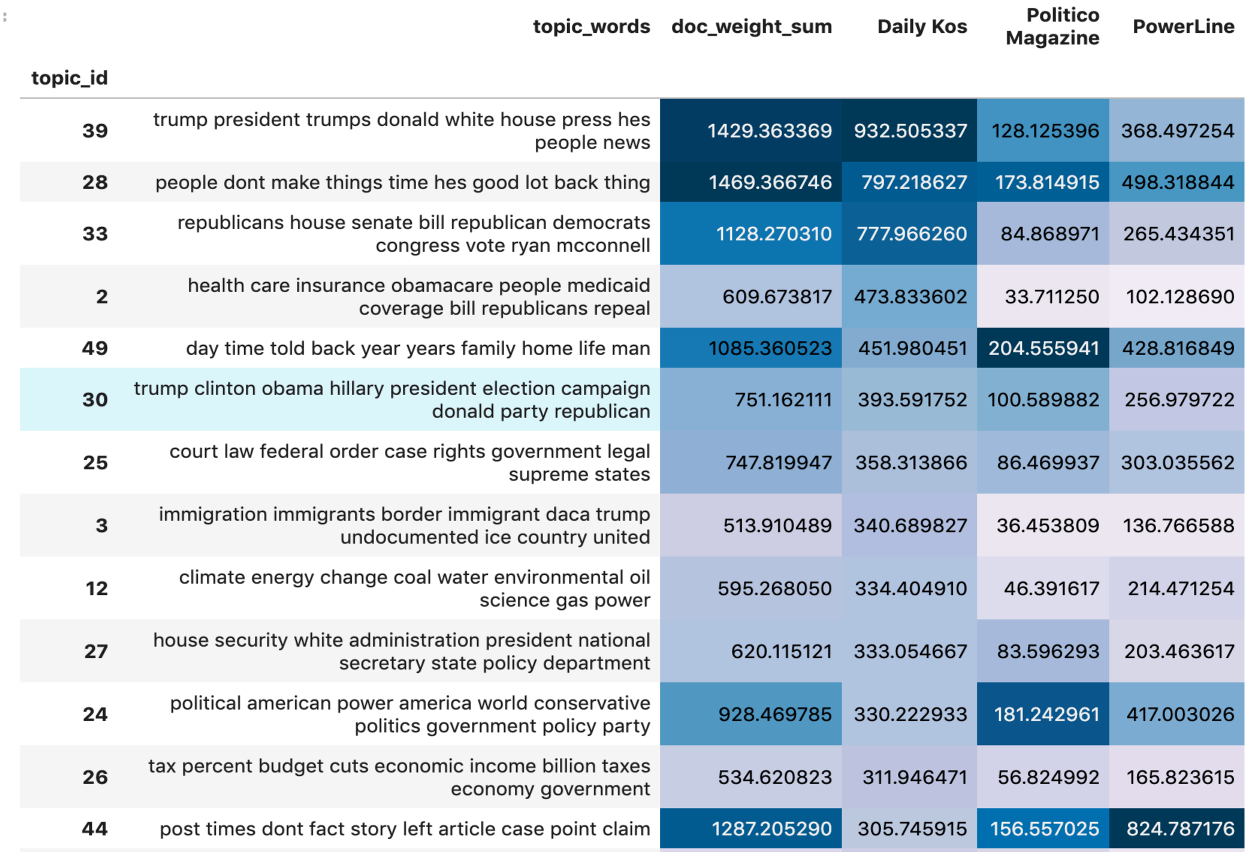

27: house security white administration president national secretary state policy department
18: president nixon kennedy reagan bush war history house john johnson
0: impeachment president trump house democrats ukraine committee trumps senate investigation
48: investigation comey sessions trump mueller attorney general fbi justice president
15: intelligence fbi information committee report investigation cia dossier campaign nunes
39: trump president trumps donald white house press hes people news
30: trump clinton obama hillary president election campaign donald party republican
38: trump russian campaign flynn manafort meeting russia kushner cohen mueller
36: north nuclear china korea trump trade united states deal chinese
34: iran israel iranian saudi israeli state president deal trump east
10: war military iraq syria isis attack forces force troops american
37: russia russian putin trump foreign president american russians states vladimir
45: state primary gop rep democratic gov seat run sen year
43: state republican virginia scott moore governor ossoff county election race
14: republican sen democrat race senate rep million poll democratic trump
9: state voting states vote election voter voters republicans republican court
33: republicans house senate bill republican democrats congress vote ryan mcconnell
32: democrats democratic district trump republican republicans house seats state gop

Figure 6 Part of the Dendrogram of Topics

The topic models, compared to the other methods, were the most revealing in terms of the priorities for each news source. In the table below, we can see that Daily Kos discussed Trump and the White House far more than the other two publications, where Powerline had many more stories about the media (topic 44).



	topic_words	doc_weight_sum	Daily Kos	Politico Magazine	PowerLine
topic_id					
39	trump president trumps donald white house press hes people news	1429.363369	932.505337	128.125396	368.497254
28	people dont make things time hes good lot back thing	1469.366746	797.218627	173.814915	498.318844
33	republicans house senate bill republican democrats congress vote ryan mcconnell	1128.270310	777.966260	84.868971	265.434351
2	health care insurance obamacare people medicaid coverage bill republicans repeal	609.673817	473.833602	33.711250	102.128690
49	day time told back year years family home life man	1085.360523	451.980451	204.555941	428.816849
30	trump clinton obama hillary president election campaign donald party republican	751.162111	393.591752	100.589882	256.979722
25	court law federal order case rights government legal supreme states	747.819947	358.313866	86.469937	303.035562
3	immigration immigrants border immigrant daca trump undocumented ice country united	513.910489	340.689827	36.453809	136.766588
12	climate energy change coal water environmental oil science gas power	595.268050	334.404910	46.391617	214.471254
27	house security white administration president national secretary state policy department	620.115121	333.054667	83.596293	203.463617
24	political american power america world conservative politics government policy party	928.469785	330.222933	181.242961	417.003026
26	tax percent budget cuts economic income billion taxes economy government	534.620823	311.946471	56.824992	165.823615
44	post times dont fact story left article case point claim	1287.205290	305.745915	156.557025	824.787176

Figure 7 Portion of table of document weight sums for each publication. See notebook for full table.

An interactive chart in the notebook also reveals that Daily Kos talked much more than Powerline about elections, Congress, and health care. Powerline discussed the media far more than any other topic. I compared these two because of the concentrations of their topics. Politico's coverage is more widespread. The chart also shows that for each publication's favored topic, it spent two to three times as much coverage (or more) than other topics. We can see this by the clustering of most topics in the bottom left of the chart.

Conclusion and Further Research

Most of these findings aren't very surprising, taking into account differences between conservative and progressive press. However, it's interesting and validating that the models were able to pick up on these differences accurately, and I suspect that if I showed the topic models to the publications' editors, they might be surprised at how many of their articles are focused on one or two topics.

Further analysis would require more time and probably more computing power, but it might be interesting to analyze the narrative structure of the articles for each source over time. It's not clear whether this would be insightful, since the sentiment analysis was inconclusive, but it would be instructive to see if the narrative aligned well with important national and world events.

More interesting would be using a similar but larger corpus to train a classification model. Using Naïve Bayes or some other method, we could train a model using sites like Powerline and Daily Kos that are partisan as their *raison d'etre*, then use the model to classify more centrist or 'neutral' news sources such as The Economist, the New York Times, or the Financial Times. These sources are not partisan but by comparing topics and topic weights, we could see if these sources spend more time discussing issues of more concern to a conservative or progressive audience. This would have to be done over a substantial amount of time, perhaps ten years or so, because partisan sites have a tendency to downplay stories that are unfavorable to their preferred president or congressional majority, e.g. Powerline’s relatively less coverage of Trump’s impeachment.

Appendix: Explanation of Tables

Note that the Jupyter Notebooks are numbered in the order that the tasks were done. PCA, word embeddings, sentiment analysis, and topic models can be looked at in any order.

TOKEN.csv – The full token table taken from the entire corpus
TOKEN2.csv – The reduced token table, with stop words and low frequency/entropy words removed
VOCAB.csv – Full vocabulary table, with all corpus words
VOCAB2.csv – Reduced vocabulary table with words used in models, includes more features and stats such as embeddings, indexed by term id
PCA.csv – Table of principal components with documents
PCA_LOADINGS.csv – table of PCA loadings, indexed by term id
SENTIMENT_DOC.csv – Sentiment polarity for each document
SENTIMENT_VOCAB.csv – Sentiment values for vocabulary, indexed by term id
TOPIC_TERMS.csv – Table of topics and term weights, indexed by term id
DOC_TOPIC_CONCENTRATION.csv – table of concentrations of topics in each document

Other csv files such as the scraped articles can be found on my UVa Box account, see manifest. For convenience I omitted the TFIDF table because of its size (about 20GB), but I can submit that by request.