Measuring Disagreement Among Media Outlets Through Twitter and Text Analysis

David Landay

Data Science 1
dlanday
University of Vermont

December 13, 2017

Abstract

Under the assumption that media outlets represent a particular political ideology, this work aims to measure the amount of disagreement among outlets by comparing words contained in tweets and newspaper articles published by seleced entities. With a keyword query of "Trump", we scrape tweets from four news agencies for 30 days in the past. We examine tweets and articles from two of the four agencies voted most representative of the political extremes, on the "left" and "right", as identified by a Pew Research study. The twitter handles selected for this project include @nytpolitics (New York Times politics), @GaurdianUS (The Guardian US), @foxnewspolitics (Fox News politics), and @BreitbartNews (Breitbart News). For each type of text, tweet or article, we construct a word corpus for each individual news outlet. We extract the a set of unique words across all corpera, and build probability distributions for each word in the unique set. We then use the Bhattacharyya distance formula, as outlined in (*Zanardo* [1]), to measure the overall disagreement amongst pairs of agencies. We compare the amount of disagreement among twitter data, to the amount of disagreement found in news articles. Through comparisons of disagreement, we show which news agencies disagree most and whether words in tweets show greater disagreement than words found in articles,among the different groups. Political extremes are then identified by the amount of disagreement exhibited.

1 Methods

1.1 Data Retrieval

A very time intensive part of this project was mining and cleaning the data. Despite queried searches, it was often necessary to manually examine the tweets that were extracted. I focused on the ”politics specific” handles associated with each news agency. For instance, The New York Times owns the twitter handles @nytimes, @nytpolitics, @nytimestech, etc... because each one is treated as a category of the news. By narrowing my focus, I was able to eliminate erroneous tweets from corrupting the data. In this way, I was attempting to ensure that only words relating to the search query, ”Trump”, were considered. In addition, certain professional criteria that news agencies adopt for posting content to twitter helped with the data cleaning process. For example, almost always was it the case that a tweet included a url to an article referenced in the tweet. Article urls were systematically the last element of a tweet, which made it easy to extract. By extracting the link, I was simultaneously able to clean the data and gain access to a secondary data set of full-length news articles.

The extraction of data involved two main python libraries: the tweepy module, which allowed me to interact with Twitter's REST and Search API's, and newspaper.py, a brilliant web scraper designed to efficiently and cleanly extract interesting parts of online news articles. Accessing the API was a multi-step process involving registering a Twitter App, generating authentication keys, and reading the tweepy docs. To maximize the amount of data at my disposal, I applied for a Twitter Premium dev account, allowing me access to tweets older than one week. A Premium/sandbox account allowed for a month's worth (in some strange cases more) of data retrieval. However, I was still confined to the same rate limits of other devs.

The inherent JSON structure of tweets meant that I could easily write to and pull from new-line separated JSON files. To expedite the data retrieval process, I used the Vermont Advanced Computing Core (VACC). Rate limits and size limits limited me to extract just over 3200 tweets per account.

Beyond traditional data cleaning, and despite my best efforts to expunge biases from the data, it is worth noting that tweets from @GuardianUS often deviated from the topic defined by the search query. For example, many references to soccer, the MLS, Champions League, and other sports were included in The Guardian's corpus. I was short of time, so I was unable to develop a method for cleaning this data, but assuming I performed my analysis correctly, and my results reflect

the true relationships among political ideologies, then the bias introduced by these tweets did little to hinder an expected result.

1.2 Methods for Analysis

To address the question of measuring disagreement among news outlets, I was inspired by the structure outlined by Zanardo in, *How To Measure Disagreement?*, who formulates a set of axioms for measuring the disagreement of two opinions. In an example provided within the paper, a set of world-states -In his case, each state is a market option- is laid out. Each market analyzer, Ann, Bob, and Carl are asked the probability of the success of each option. Zanardo claims that the probability each person assigned to a state is the opinion they hold of that option succeeding. To measure disagreement among each individual, a distance formula is used to compute the "physical" distance between the probability distributions. In an attempt to work from Zanardo's outline, I selected the Bhattacharyya formula for computing the distance of a probability distribution of states, because it conforms to all of the 6 axioms for distance formulas. In my project, I consider each tweet or article (document in a corpus) a state, and for each unique word in the set of states, the probability of that word is calculated. The probability distribution of words in a corpus is then sent to the Bhattacharyya formula against the probability distribution of another corpus. The goal here is that two groups which exhibit the most disagreement will represent the two ideological extremes.

2 Results

After cleaning the tweets, Term Frequency - Inverse Document Frequency (tf-idf) was performed over all corpera as a form of feature selection. A unique set of words and their scores were created from the results of tf-idf and ordered from greatest to least value tf-idf score. For each word, a conditional probability was calculated of the form:

$$P(W|O) = \frac{P(O|W)P(W)}{P(O)}$$

for which:
$\rightarrow$ $P(W|O)$ is the posterior probability of a word given a news outlet has tweeted.
$\rightarrow$ $P(O|W)$ is the likelihood estimate of a news outlet tweeting given a word is in a tweet.
$\rightarrow$ $P(W)$ is the prior estimate of a word is in a tweet. $\rightarrow$ $P(O)$ is the marginal likelihood constant which is 1/4 in this case

	NYT	Guard	Fox	Breit
NYT	0	8.088	8.122	8.426
Guard	8.088	0	8.483	8.388
Fox	8.122	8.145	0	8.483
Breit	8.426	8.388	8.483	0

Disagreement Matrix: showing disagreement amongst pairwise news groups

Above is the disagreement matrix amongst all news outlets. It shows, based on the words among the corpera, that Breitbart news disagrees most with all other news outlets. Surprisingly, it Fox News and Breitbart disagree the most. Assuming these were calculated correctly, these results show (more or less) what we expected to see. Under our initial assumptions, the results show that Breitbart, being a self-proclaimed "alt-right" news organization, is one of the political extremes. As one might suspect, we could scale the experiment by comparing a larger number of news outlets. The resolution of the political spectrum we are trying to build will improve in that we will discover more intermediary news agencies. That is, we can compare more news groups and determine which groups disagree most. The results of this test are flawed in that Fox News is considered the other extreme, which is incorrect.

Figure 1: Nytimes words sized by relevance to the corpus

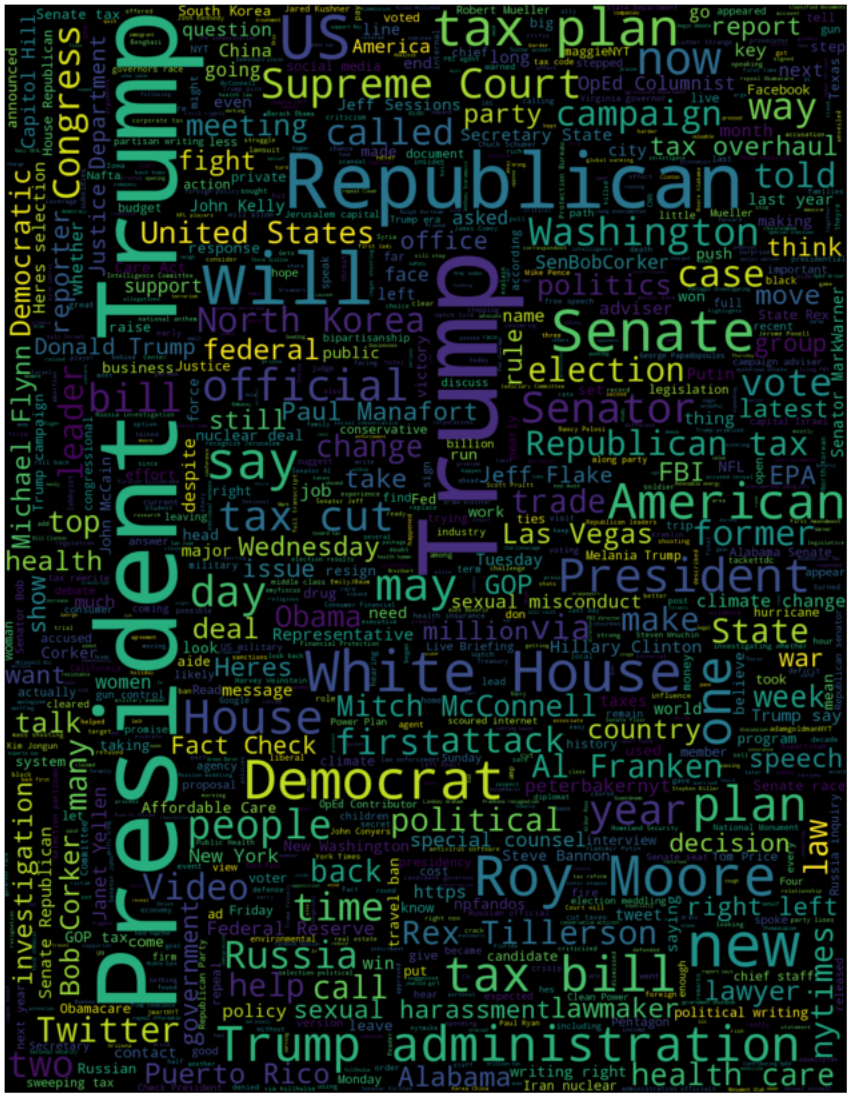


Figure 2: GuardianUS words sized by relevance to the corpus

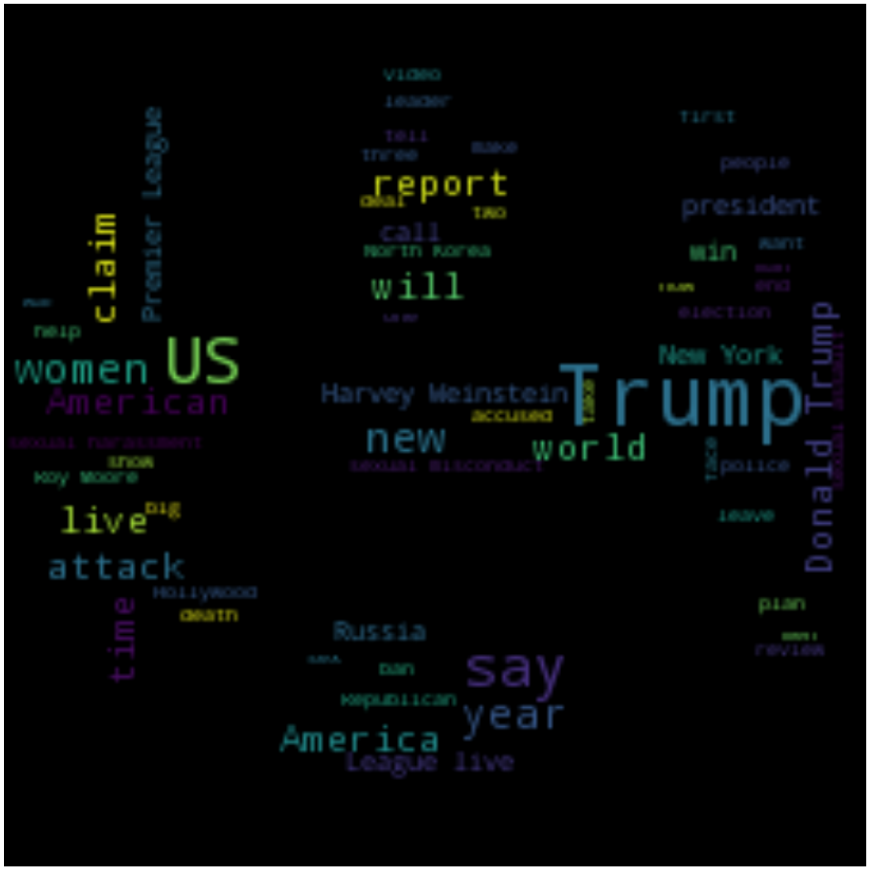


Figure 3: Fox News words sized by relevance to the corpus

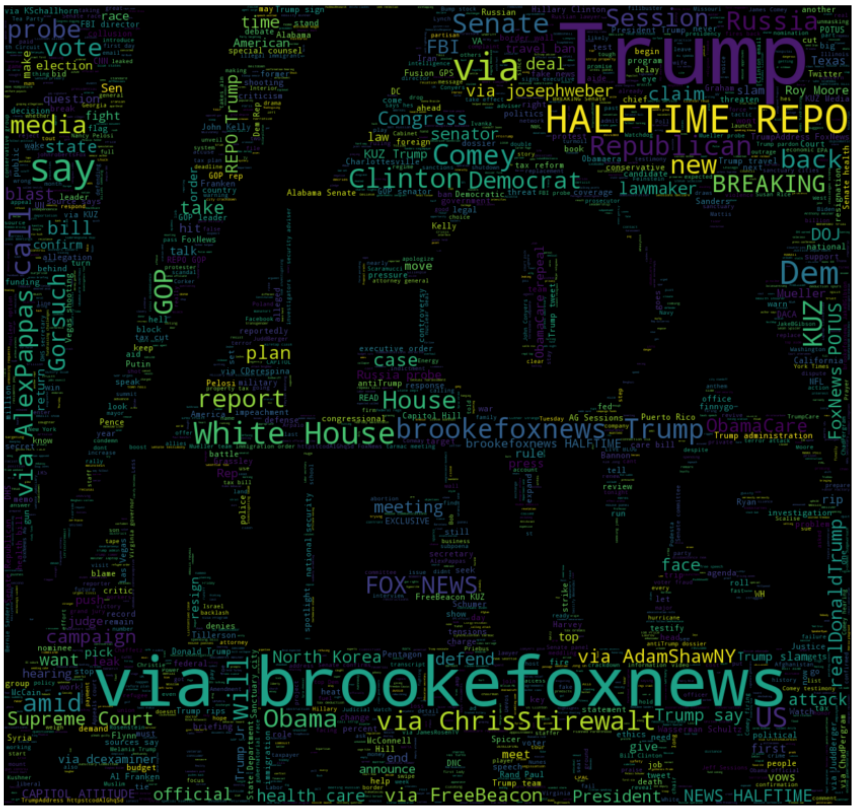
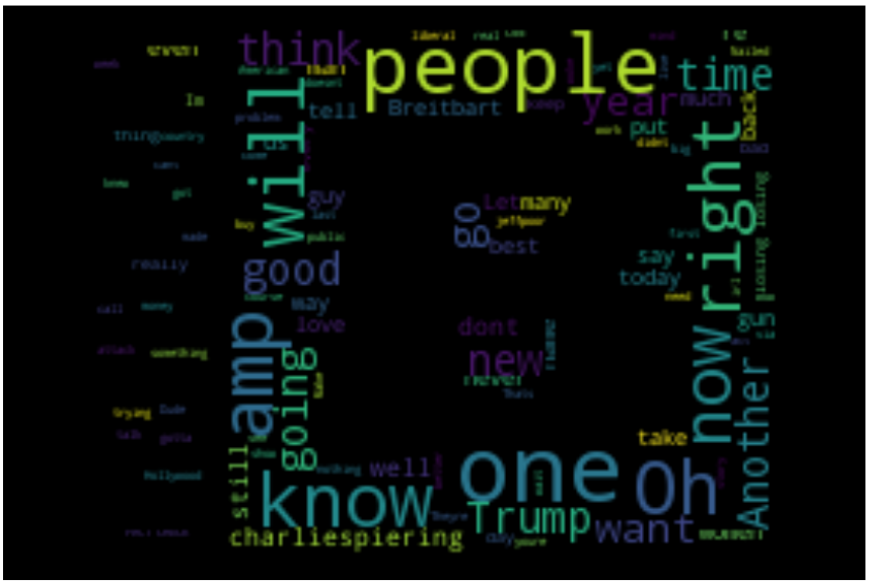


Figure 4: BreitBart News words sized by relevance to the corpus



The above are a series of wordcloud plots. They highlight the importance of words in a corpus, based on a tf-idf score. In the code, you will notice that I performed my own version of tf-idf for scoring words in each corpus. What these plots confirm is that we are constructing or corpera correctly from the query search.

I was also interested in which news outlet gets retweeted most often. Below are a series of time series plots describing news outlets that were retweeted most often:

Figure 5: Time Series of Retweets

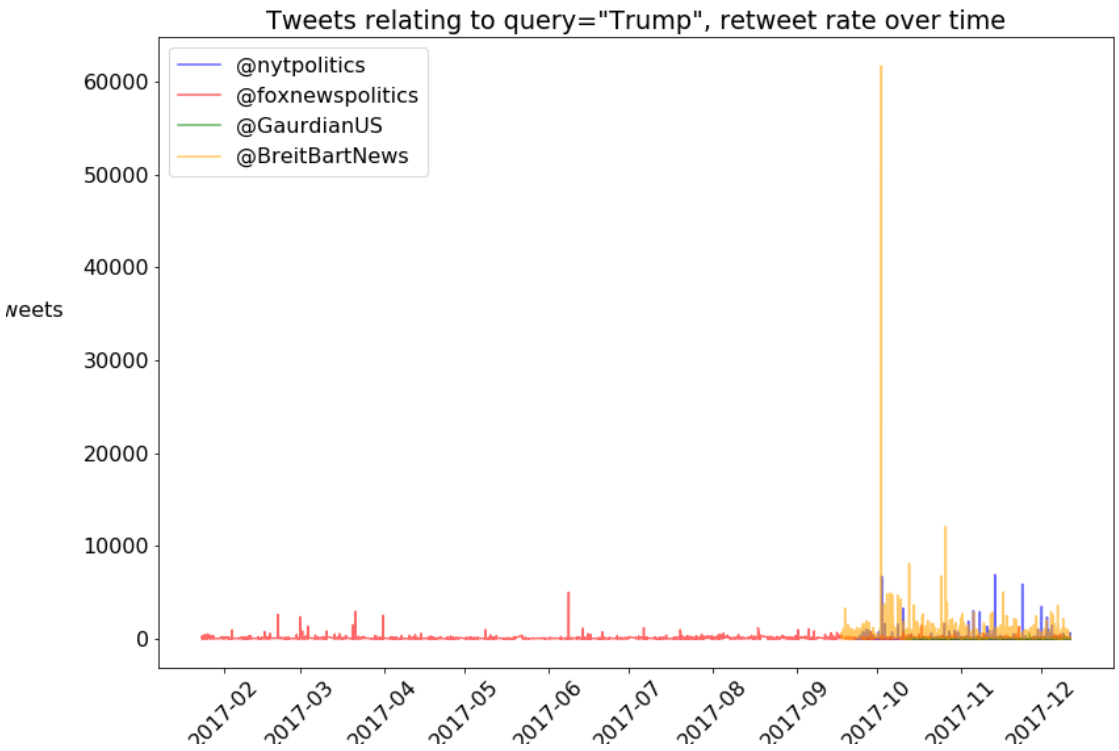


Figure 6: Time Series of Retweets [scaled window]

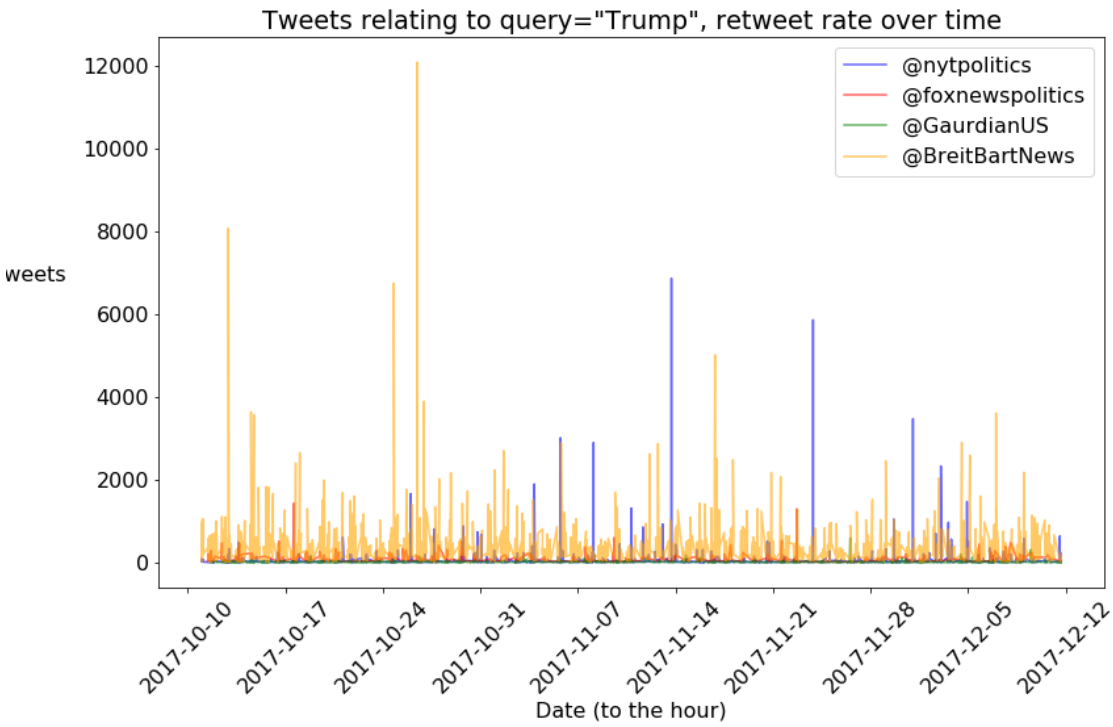
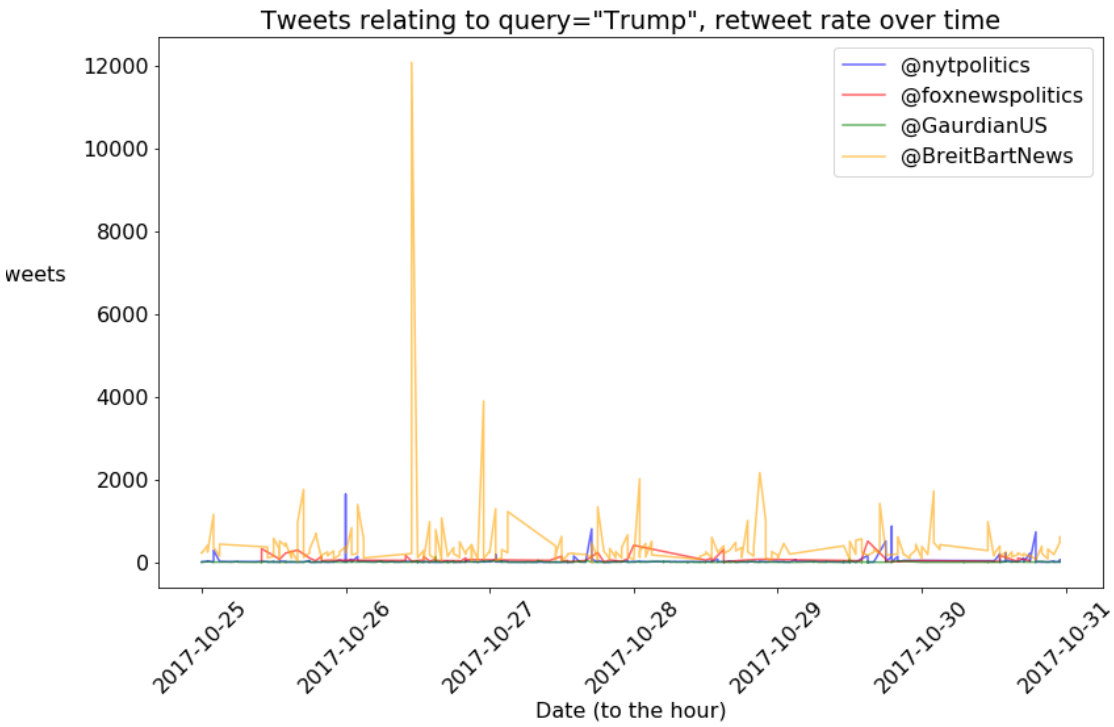


Figure 7: Time Series of Retweets [scaled window]



3 Conclusions and Future

Based on the initial assumptions made about news outlets representing a population, we can see that the method we constructed captures some trends, but is not perfect. A reasonable suspicion is that more data cleaning needs to be done to ensure an accurate test. Also, I would have liked to have implemented augmented tf-idf, which accounts for the variance in document length. For agencies like Breitbart, this may better represent sentiment among words because the average length of tweets was low, relative to that of the other groups. But, perhaps this is why Breitbart has a higher re-tweet rate than any other news agency. Another speculation is that Breitbart truly represents a unique population of conservatives. Perhaps Breitbart appeals to people who spend more time on the computer, or phone, whereby they can spread Breitbart articles more readily than other outlets can afford. I would like to investigate this trend in a future study.

As I was pressed for time, I was un-able to compare the disagreement scores for web articles. My cpu was struggling to parse the amount of text available. In the

code, I have demonstrated that I know how to clean, and evaluate the article texts. I have even constructed my internal functions such that, I can perform the same analysis on articles as I did for tweets. The original intent of this project was to develop a visualization of the political spectrum, given the extremes of words. As of now, disagreement is purely a distance between two corpera. Re-assessing, I will probably need to use further sentiment analysis to actually scale words.

This project taught me much about NLP and the structure of language. I come a way with a better appreciation for text as a source of data, and the dilemmas associated with processing large quantities of it.
I wish to continue this investigation to better understand the disconnect between political parties.

4 Links

- http://www.journalism.org/2014/10/21/political-polarization-media-habits/
- https://newsapi.org
- https://www.crummy.com/software/BeautifulSoup/bs4/doc/
- https://marcobonzanini.com/2015/03/02/mining-twitter-data-with-python-part-1/
- https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4574198/

References

[1] Zanardo, Enrico *How To Measure Disagreement*. Working Paper, **38**,Columbia University, (2017).

[2] Sylwester, Karolina, and Matthew Purver. *Twitter Language Use Reflects Psychological Differences between Democrats and Republicans* Ed. Christopher M. Danforth. PLoS ONE 10.9 (2015): e0137422. PMC. Web. 21 Nov. 2017.