Word Frequency Comparisons between George W. Bush and Barack Obama's Top 50 Speeches

Winnie Yan

CUNY Graduate Center

Introduction

For my term project, I initially thought to count the most frequent words used in the top 100 speeches on the American Rhetoric website. After discussing my idea with Kyle, he suggested I instead count the most frequent words used in the top 50 speeches of two different categories, and then use log-odds ratios to compare word frequencies between the corpora. For my two categories, I chose the Top 50 speeches by President George W. Bush and the Top 50 speeches by President Barack Obama. For the word frequency comparison, I was to use a third corpus as a "background" or "control" corpus. The intended goal of the project was to use log-odds ratios to find the frequency scores of words present in all three corpora.

Methodology

Part 1: Scraping

The first part of my project required scraping the website for the speeches. I found a web-scraper for the American Rhetoric website that someone on GitHub had previously built in 2015. However, some changes had to be made to the code to account for changes made to the website, such as the use of .html as well as .htm pages. I also had to change the source URL because the original code only scraped the page for Obama speeches but I also needed the page for Bush speeches. Additionally, I changed the amount of speeches retrieved from each page to 50, and used UTF-16 encoding when saving the files. I also deleted part of the original code that created an Excel directory of the scraped speeches, as that was unnecessary for my project.

Since I was using existing code, I did not encounter many issues in this part of the project. When I first started writing the code, I began by only trying to scrape five speeches from the Bush website so I could identify and correct errors faster than if I had tried to do all 50 at once. After I finished scraping the speeches, I realized I wanted the Bush speeches and the

Obama speeches stored separately so I looked up how to create a new directory and had the speeches written to files within their respective directories.

Part 2: Counting

The second part of my project involved running word counts on the collected corpora. In our final practicum, Jon showed us how to use NLTK to and collections.Counter() to sentence tokenize, word tokenize, and count the words in each file. He also showed us how to use most_common() to organize the counted words from most frequent to least frequent. I found that the NLTK library has a set of stop words that I could use to eliminate extraneous words such as "the", "with", etc. In writing Part 2, I decided to break the code up into functions to decrease the number of lines of code, since I had to run everything once for the Bush speeches, and once for the Obama speeches.

My first function was written to get the word count for each set of speeches. I had the function read through all the files within a given directory, casefold the strings, sentence tokenize, and word tokenize. Then, for every word present, if the word was not one of the stop words and it consisted only of alphabetical characters, then it was added to the speech count list and the counter would provide the number of occurrences of the word. I included [.isalpha] because I realized commas and periods were still showing up when I used most_common(). I also included print statements to indicate when the file reading began and when it was completed, and a count of the total number of tokens across all the speeches in each directory.

The second function was designed to create another new directory that contained a single file where the previously obtained wordcount was stored. The purpose of creating the new directory was because my wordcount file was initially being stored into the same directory as the speeches, which interfered with the counter if I ran the code again without deleting the previous

word count file. Since most_common() returned information from a counter, I had to write to the new file using separate write statements to include the word, a tab, the count of the word, and a linebreak.

The third function was the main function. In the main function, I had one section for each set of corpus data. The Bush and Obama sections were identical, where I called the previous functions to count words from the respective directories and then create new directories with the word counts saved in a .txt file. The last section of the function was to create a word count for the control corpus. I chose all the files in the NLTK Reuters Library as the control corpus for my word frequency comparisons. Originally, I used the KJV Bible but I realized the words in that corpus were unnatural to the context of the speeches I used and my data became skewed. I also decided not to use the news corpus provided by Kyle because it was collected in 2017, after most or all of Bush and Obama's speeches. I figured the data from the Reuters Library would be more in line with the context of the speeches than the KJV Bible or the 2017 news corpus. For all the files in the library, I added it to a variable, sentence and word tokenized it, and then counted the frequency of each word that occurred. Then I also created a new directory with a file containing the word frequencies, and printed the total number of tokens in the corpus. I finished Part 2 by calling the main function.

Part 3: Log-Odds Ratios & Comparing Word-Frequencies

For the third part of my project, Kyle provided me with lecture slides for log-odds ratios, as well as the snippet of code for computing log-odds. I decided to change the args. paths to the actual file paths created in the previous parts of the project instead of having to write them in the console. From what I understood of the slides and the code, there was a section for [if not raw.arg] that computed z-score. However, with this portion in the code, I found my log-odds

ratios skewed very heavily towards the words found in the control corpus instead of the words found in the speeches. Based on this, I decided to remove that section of code so the log-odds ratios list was a better reflection of the words found in the speech corpora. Instead of printing the log-odds ratios, I chose to write them into a file under a new directory as well, to keep the information separate.

Results

The total number of tokens for all 50 Bush speeches was 54,487. The total number of tokens for all 50 Obama speeches was 40,801. The total number of tokens in the Reuters library corpus was 838,830.

In the Bush speeches, the top 20 words and their frequencies were: [“thank 170”, “would 170”, “peace 168”, “make 165”, “americans 164”, “terrorists 156”, “free 154”, “today 146”, “also 145”, “yet 143”, “military 140”, “iraqi 138”, “regime 134”, “years 131”, “need 131”, “life 127”, “citizens 125”, “children 120”, “way 117”, “threat 117”].

In the Obama speeches, the top 20 words and their frequencies were: [“us 340”, “people 293”, “america 263”, “new 260”, “american 232”, “time 225”, “one 215”, “know 203”, “country 200”, “work 197”, “must 193”, “world 187”, “president 185”, “war 175”, “need 166”, “make 164”, “jobs 151”, “americans 148”, “many 147”, “nation 144].

From the Reuters corpus, the top 20 words and their frequencies were: [“said 25381”, “mln 18598”, “vs 14332”, “dlrs 12329”, “pct 9771”, “lt 8696”, “cts 8308”, “net 6986”, “year 6687”, “billion 5809”, “loss 5115”, “would 4685”, “company 4593”, “shr 4131”, “inc 3919”, “bank 3598”, “corp 3258”, “last 3227”, “oil 3193”, “share 3084”].

After calculating the log-odds ratios, the top 20 words and their ratios were: [“nations 0.4478819461451784”, “iraq 0.4470637950148535”, “freedom 0.44674399268056275”, “every

0.446654139238559", "world 0.446552104973871", "peace 0.4460854619506715 6", "people 0.446007898490657 7", "regime 0.4459422151417969", "weapons 0.44594131473524756", "america 0.445908032508501 4", "free 0.44568234001340734", "united 0.445664593507054", "iraqi 0.4454225948838304", "terrorists 0.44515247110308387", "yet 0.44507549878224006", "citizens 0.445055933790565 3", "saddam 0.4450471358444261", "great 0.4449030431735559", "security 0.44490049709116963", "good 0.44489792500295255"].

Discussion & Conclusion

As observed in the data, the Reuters corpus used a lot of abbreviations and terms that were not present in the speech corpora. There was also a lot of overlap between the most frequently used words between Bush's speeches and Obama's speeches, although they did not occur at the same frequencies. The top 20 words in the log-odds ratios reflected both speech corpora, but more from the Bush speeches than the Obama speeches, and had little overlap with the Reuters library corpus. This indicates that Bush and Obama used similar words in their speeches but the Reuters corpus did not. The heavier leaning towards the words in the Bush speeches may be due to the Bush speech corpus being larger than the Obama speech corpus. Alternatively, many words in the top-20 of the log-odds ratios were related to the Iraq war, which was heavily discussed by Bush and may have been discussed in the Reuters library, but was not significantly discussed by Obama. According the top 20 words in the log-odds ratios, the most common topics relate to the Iraq war, and include words like "peace", "freedom", "weapons", etc., which overlaps with Bush's speech rhetoric. Obama's speech rhetoric reflects more on American unity and employment with words like "work", "make", "jobs", etc.

Sources

Speeches retrieved from: https://www.americanrhetoric.com/

Web scraping code: https://github.com/yimihua2013/Mining-the-Web_Python/blob/master/ObamaSpeech.py

Lecture slides on log-odds ratios: http://wellformedness.com/courses/LING83600/PDFs/lecture02.pdf

Dec. 10th, 2019 Practicum with Jon – NLTK, collections counter, most common

Creating a new directory: https://stackabuse.com/creating-and-deleting-directories-with-python/

Going through files in a directory: https://stackoverflow.com/questions/3207219/how-do-i-list-all-files-of-a-directory

How to remove stop words: https://www.geeksforgeeks.org/removing-stop-words-nltk-python/

NLTK corpus libraries: https://www.nltk.org/book/ch02.html