

Design Process/Changes Made

During the implementation phase of working on this project, I ran into many issues and challenges that I had to overcome in order to get the project to where it is today. Right off the bat with data collection, I originally started off on the wrong foot. I had planned to scrape the text for each of the presidential speeches from the [Miller Center](#) website using the BeautifulSoup python library combined with the Selenium python library for page navigation. This, I quickly discovered would be a huge barrier to my progress. The reason for that being that the website is dynamically loaded, and in a manner that made it particularly annoying to navigate with a headless browser using Selenium. Luckily I didn't give up on this project idea there, because upon further inspection of the website they actually provided a decent API that allowed me to collect all of the presidential speech data that I required.

From that point, I was able to move into the data cleaning phase, where I cleaned up the poorly-formatted json file and saved it as a csv file so that I would have a bit of an easier time working with it and loading it within python as a dataframe. The next issue I discovered was that I would have to find a dataset containing the information on which president was a part of which party and join that data to the existing data as a new column if I wanted to be able to not compare individual presidents, but instead compare the parties that they were members of. This was an easy enough fix, but I ran into several issues in doing so originally because the formatting from the json file I received from the Miller Center API was very poor. Once I finally had all of my data cleaned up (or so I thought), I began trying to feed my data to the scattertext model I created. I quickly discovered that scattertext only allows for you to directly compare two categories at a time using the scatter-plot it created. This presented an issue for me, since there were many political parties whose presidential speeches I wanted to compare together. I decided that the best method to utilize this library for my purposes was to separate charts into time periods depending on the era of US politics they occurred during. I also decided that it would be interesting to compare each party in a time-period against all other parties in that same time-period. That is how I eventually came to the resolution to use the model in the manner I did for this project. I then had to go back to the drawing board and split the speeches up based on the time-period in which they were given. Once this was done, I was able to convert each CSV representing an era in US politics into a scattertext chart for each major party that held presidential office during that time. This still left an issue with the scattertext chart, that I tried for quite a while to solve to not much luck: loading times. The interactive charts created by the scattertext model load incredibly slowly in the browser, and on charts that were made from a corpus (groups of speeches) of many speeches they would cause the browser to crash. I tried decreasing the model's size, decreasing its complexity, increasing how many charts I was creating so there would be less to render on each page, and many other solutions. I finally got it to the point where it was working, with a short ~10 second wait while the page loads in.

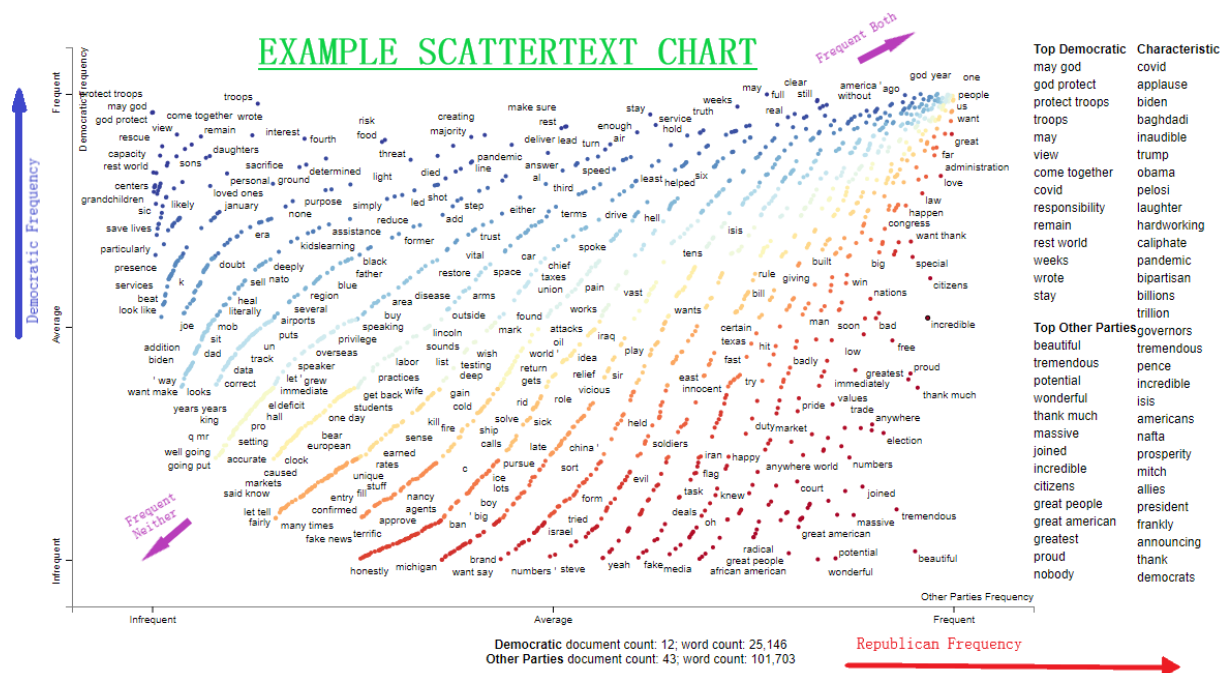
The last and final hurdle I had to clear was figuring out how to present the 20 charts that were created in this manner in an easy to understand way, while maintaining the interactivity provided by the scattertext model. In the end I decided to create an html file that would work like a gallery so that you can see all the charts in one location, have them properly organized by the time-period they were given, as well as have them link to the fully interactive version of each one in a new tab from there.

Visualization Summary

My visualization tells much of the story of American politics in one place. All of the presidential speeches given in the United States have been included in this model. Say for example, you would like to compare the Democratic Party and Republican Party of the Regan era with the modern parties of today; with this visualization that would be incredibly easy to do. It is the job of the presidential speech writers to distill all of the talking points and debate topics that have been popular in the mainstream into their speeches. To put it another way: all of our nations top issues of the day are the ones that are brought to the table when it comes down to the writing of presidential speeches.

The main results I was interested in seeing when going to create this visualization were found by comparing the hawkishness of the two modern mainstream parties, with the parties during the 40s (during the 5th Party System of United States Politics). The results I found were pretty interesting; I discovered that the Republican party has really not gotten as far away from it's roots with it's modern presidential speeches as many might would believe. The China hawkishness seen in their speeches, and the framing of speaking of their economic policies as being "pro-business" really hasn't changed since the 40's. The democratic party, on the other hand, seems to have had a lot more changes in the way that they frame political issues through their presidential speeches. During the 40's, it would seem that they used a lot more overtly pro-consumer language in their speeches, whereas in more recent years they seem to tow the line between using language that is either pro-business or pro-consumer much more.

Visualization Access



My visualization, as previously stated can be accessed primarily through the HTML page I set up to act as the “home page” of the project, which is labeled ‘*US_Presidential_Speeches.html*’. Once this page loads into your browser, from top to bottom as you go down the dates of the speeches in each era get more current. In each category you can select the party you would like to look at, and click on it to load the model in a new window. Once the model loads in (keep in mind it will be a tad slow), you will see a scatter chart with the party you are directly looking at on the Y-axis, and the parties that you are comparing it against during that era on the X-axis. As you go further up and to the right, you will see words that appear more frequently for all parties during that era. As you look further downward and leftward, you will see words that are used less frequently within all parties during that era. In the top left corner of the chart you will see words that are almost exclusively by the main party that you are looking at within that time-period, and in the bottom right corner of the chart you will see words that are predominantly used by parties aside from the one you are looking at in that era. For interactivity, you can hover over words to see the exact frequency that they are used by each group, and you can click on a word in order to have the context of the word withing the speeches that it was found shown to you along the bottom of the screen. The parts of speeches that it shows when this happens have had their “stopwords” (words such as ‘a’, ‘is’, ‘the’) stripped from them so that it reduces loading times. You are still able to get a good idea of the context that each word is being used with in each specific speech using this interactive methodology. The last bit of interactivity that you can use with the scattertext chart is the search function. The search function works in a similar manner to clicking on a word you can see on the chart, however, the main difference being that you can search for any word you would like and see it’s context within the model regardless of how frequently it showed up. What you can directly see on the scattertext chart normally are words used more than ~8 times within the presidential speeches of each era.

Design Decisions

The biggest design decision when it came to the project visuals was deciding how I would like to present the large amount of data I had from this project, since I had created 20 charts that could be used in similar methods to the one I laid out in the Visualization Summary section of this final project writeup. In the end I decided to use the principle of small multiples to display each of the charts side-by-side so that the viewer would be able to easily navigate through each one, and find and directly compare each one with the others. I color-coded the title and subtitle regions on the html page I created so that each era would have a clear dividing line between it and the next. I also upscaled the charts from the Fourth Party System onward since there was only two major parties in each time- period and it made the charts on the main page a lot more legible. I also mulled over with making the design decision to not leave the background simple and plain, but in the end I decided against it. There was already a lot of color being provided by the scattertext chart, so I chose to leave the background white so that the colorful points would pop a lot more and be easier to read.

Discussion of Future Changes

I had to settle for not letting good be the enemy of great in the case of loading times. If I had more time to work on the project further, I would like to try to further optimize the scattertext model. The issue of loading times likely occurs due to some inefficiencies of the model itself when dealing with as wide a body of text as I used for this project. I optimized the model itself by increasing the number of

times a word had to be present in the model, which lead to the removal of points from each scattertext chart and helped to boost the speeds of the interactive webpage that each chart renders to. This increased the overall speed of the model's interactive segments a good deal. I originally had it set to require a minimum of 5 occurrences to include per word from each grouping of speeches. I tested out different numbers for that until I arrived at using 8 as the minimum requirement for occurrences. In retrospect, having finished the project, I likely could have further optimized the model by bumping that number up a little bit more. I would also like to try comparing parties during other time periods, or possibly limiting the scope to compare individual presidents to the rest of the presidents.