Project Write-Up and Reflection

Prava Dhulipalla

23 February 2017

Mini-Project 3: Text Mining and Analysis

Software Design Spring 2017

Project Overview

Since I was interested in analyzing inaugural speeches, my data source was an html page that linked to all the speeches. Because part of this project is to learn data mining, for my first deliverable, I used the Olin Wikipedia page and generated a list of the top 50 words and a corresponding word cloud. For my second deliverable, I simply copy and pasted the text into .txt files, and had my file read from there. To analyse them, I used word frequency analysis - two ways. First, I wanted to see which words were used the most overall, across all the inaugural speeches. Second, I wanted to see which words were used across the most documents. I wanted a more visual representation of the data, so I created word clouds. In this process, I was hoping to learn more about dictionaries and how to use them, learn more about algorithms (like word frequency analysis), implement relevant libraries to aid with my final product, and be able to collect data both from the internet and from computer files.

Implementation

This project had three main python files. text_analysis.py had the important functions: the function that put all the words and their counts in a dictionary and the function that printed out the list of the top 50 words and their corresponding counts as well as a word cloud. data_mining.py took the content from Olin's Wikipedia page and used text_analysis.py to produce the printed list of the top 50 words and the word cloud. text_mining.py was the 'main class' that read all the inaugural addresses from the computer, and created two dictionaries for words: one that held the most used words overall, and one that held the words used by the most amount of speeches. Two print lists and word clouds were generated from this.

One major design decision I made was to not get the inaugural address data from the internet. This was done for two reasons: A) by using .txt files to hold the data, I completed my learning goal of figuring out how to call and parse data from a file on the computer (turns out, it isn't hard); and B) it seemed like a waste of time to write the code to remove all the extraneous html tags (BeautifulSoup didn't work for some reason, and the other libraries or code that I found that was supposed to remove html tags were non- or only partially functional). I didn't want to write about fifty small snippets of code, each individualized to index from the start to the end. (I suppose if I was more adept at code or had more time, I would be able to write a function or a class that would do this without needing to write about fifty snippets of code, but alas, this is not true.) I felt that the learning payoff was minimal (especially considering that I was able to snip

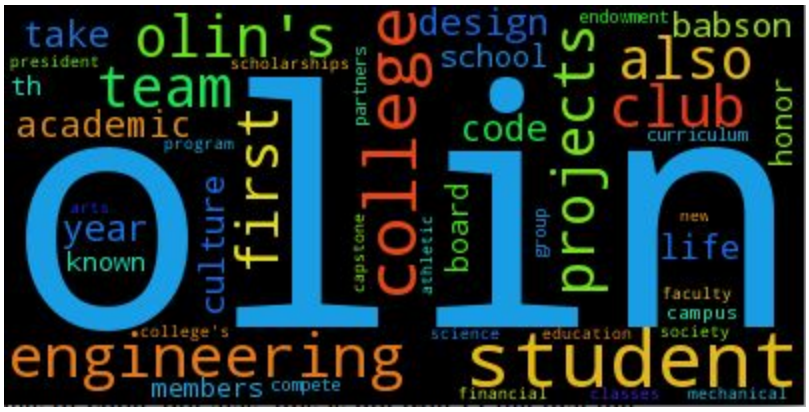
from the start to the end of the Olin Wikipedia page, so I knew I could do it). Ultimately, I feel okay with this decision.

Results

Deliverable 1: Olin Top 50 List and Word Cloud

```
Top words for Olin Wikipedia page
---------------------------------
WORD    COUNT
olin    51
college    30
students    29
engineering    18
student    14
projects    9
first    9
also    8
olin's    8
design    7
take    6
academic    6
year    6
—    6
honor    6
culture    6
board    6
school    6
life    6
babson    6
code    6
endowment    5
known    5
group    5
members    5
curriculum    5
compete    5
w    5
campus    5
partners    5
```

```
th    5
club    4
new    4
faculty    4
arts    4
classes    4
capstone    4
teams    4
scholarships    4
clubs    4
athletic    4
science    4
team    4
program    4
president    4
college's    3
education    3
mechanical    3
financial    3
society    3
```



Here is the output using the Olin Wikipedia page. Aside from a few ‘weird’ words, like ‘w’ or ‘th’ in the printed - the output is fairly expected and illuminating. ‘Olin’ is obviously going to be used a lot on the Olin Wikipedia page. Also note the tiny little ‘Babson’ in the corner - it kind of shows how Babson is actually pretty prominent figure in an Oliners’ life. There’s also ‘engineering,’ ‘design,’ ‘code,’ and ‘mechanical’ - also self-explanatory if you go to Olin. I especially like how ‘projects’ are in there, and it really shows the emphasis that Olin puts on project-based learning. All in all, nothing from this was especially surprising.

Deliverable 2: Inaugural Addresses Top 50 List and Word Cloud for Most Frequency Words and Most Frequently-used-in-speed Words

```
Words used most in all texts together
-------------------------------------
WORD    COUNT
will    908
government    568
people    568
us    478
upon    374
must    365
great    338
may    337
states    332
shall    313
world    310
country    300
every    298
nation    292
peace    253
one    251
new    247
power    234
public    226
now    222
time    215
constitution    205
united    202
nations    197
union    184
free    183
freedom    183
america    179
war    171
american    163
citizens    158
national    157
```

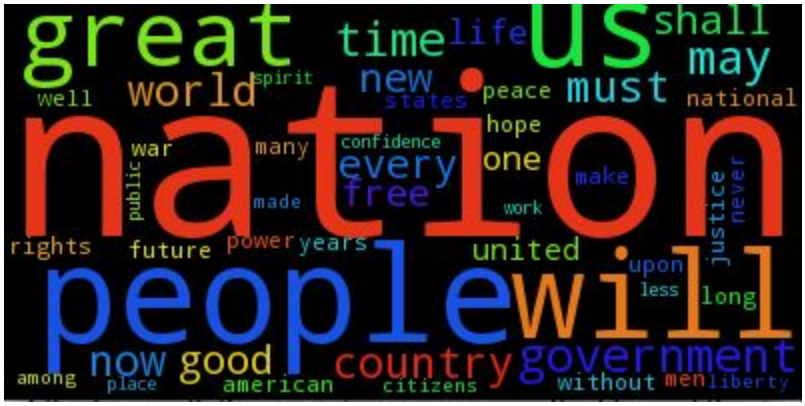
made 157
let 152
make 145
good 145
men 139
years 138
justice 138
rights 138
without 137
spirit 137
life 133
laws 131
never 129
congress 129
law 126
best 120
right 118
well 117

Words used in most texts

WORD DOCUMENTS
will 56
people 56
great 55
us 55
nation 53
government 53

may 53
time 53
now 52
world 52
country 52
must 51
every 51
nations 51
shall 50
free 49
new 49
good 49
life 48
one 48
men 47
rights 47
future 47
upon 47
united 47
power 47
peace 46
hope 46
justice 46
states 46
never 46
long 45
without 45
make 45
national 45
american 45
years 44
well 44
war 44
among 43
citizens 43
made 43
many 43
less 42
confidence 42

```
liberty    42
spirit     42
public     41
work     41
place     41
```



In the words that were used the most in general, ‘will,’ ‘government,’ and ‘people,’ are all words that appeared the most (almost 600 or more times). I knew that a lot of political words, like ‘government,’ would be used, but I wasn’t necessarily expecting ‘will’ - but I know that it makes sense. Presidents talk a lot about what they *will* do in their inaugural addresses, which is probably why it was said almost 1000 times. In the words that were used in the most amount of speeches, ‘will,’ ‘government,’ and ‘people’ all made an appearance again, as well as ‘great,’ ‘nation,’ ‘us,’ and many other words (the counts were very similar since there are only 56 speeches, so there wasn’t a great diversity in numbers, obviously). The one i will comment on is ‘great,’ a word that has been associated with Donald Trump. But it shows that before the word became somewhat commercialized in the recent election, it was genuine ideal that made presidents held to. Although this text analysis most just confirmed my suspicions, it was nice to see the words laid out in the list and in the word cloud.

Reflection

I think overall, the project was a success. It addressed the goals I wanted to meet for this project (I did learn a lot about dictionaries, algorithms, libraries, and data mining from the internet and computer files). However, if I were to continue working on this in the future, I’d start with creating a word cloud that represented the most words used and the words used across the most inaugural speeches in one word cloud: perhaps the darker the word color, the more presidents used them. Additionally, I would probably try to create my word cloud by myself so that I could address another learning goal of mine: learning more about creating graphics. I suppose I wish I did that for this deliverable, but I was told that the amount of work I had already put in was

enough. Another thing I wish I did this project that I can continue working on for other projects is working on doing unit tests, writing doc strings, writing comments, and making a project skeleton/pseudo code *before* I actually coded. I think that would have been helpful. Also, I think I should work on being more succinct in my project write-ups and reflection because this is probably *way* too much writing.

Results *[~2-3 paragraphs + figures/examples]*
Present what you accomplished:

- If you did some text analysis, what interesting things did you find? Graphs or other visualizations may be very useful here for showing your results.
- If you created a program that does something interesting (e.g. a Markov text synthesizer), be sure to provide a few interesting examples of the program's output.