Alexander Lee
Tim Woods
1-30-18
Weekly Progress Report

Dr. Hearne,

Our group has discussed the high-level aspects of our project, and has begun to experiment with using the Google Cloud Natural Language API for sentiment analysis of text.

We've identified the various components we'll need to implement in this project:

- A web scraper which retrieves the Senate's role-call vote data. These are XML files
- A web scraper which retrieves the House's role-call vote data. This is not in an easily digested format currently, so this scraper will be converting text displayed on a web page into a usable format
- An XML parser which maps the Senate data to the format used by the OpenElections project in their .csv files
- A component that maps the congressional districts in the OpenElections data to the representatives in the Congress.gov data. Could be a relational database (which might be handy for other portions of the project), or a master dictionary file
- A script to call the Google API on the text content of each ballot measure, and store the results of the sentiment analysis of that ballot measure
- A comparison engine for the sentiments of congressional districts versus their representatives.

We've looked into using the Python libraray beautifulsoup for the web scraping, and will be implementing the scrapers this week.

Tim has set up a trial account with Google Cloud services, and has done some tests to verify connecting to the Natural Language API. A combination of the analyze sentiment and analyze text features of the API seem to be the most descriptive of the handful of ballot measure he's tested.