Deliverable 1: NAACP Spark Project

- Data Collection:

This project focuses on three data sets:

- Boston Globe
- WGBH Local news
- WBUR Local news

Boston Globe data set has already been web scraped by the previous team. Until now we have finished web scraping both the WBUR and WGBH websites and stored them as csv files. Similar to the Boston globe data set, data for each year was stored separately.

- Preliminary Analysis:

Initially, we wanted to visualize the distribution of black vs non-black neighborhoods in Boston. Using the data set given to us for the neighborhoods, two different plots were generated:

- The proportion of Black Americans by zip code
- The distribution of the population of Black Americans over the map

These two maps are then used as base map such that we can overlay the results of our topic modeling for better visualization. To attempt to do this part, we created an arbitrary csv file with 5 different topics and 26 unique tags or keywords associated with that specific topic. Then we took a sample set of data from the WBUR data from 2018 and cleaned it in such a way that the output is csv file where each row is an instance of topics being mentioned and neighborhoods being mentioned in each article from the WBUR dataset. The imported new csv file was then overlayed over the previous map in order to visualize the topics. For each a component for frequency was used to make

the points larger or smaller depending on the amount of mentions that topic had for that particular neighborhood.

- Relevant Question for our project proposal:

The previous model for sentiment analysis had a temporal limitation where it needed to explore terms that were within close proximity of each other (i.e. it did not have a very good memory). To overcome this issue, we propose using a Bidirectional LSTM (Long Short Term Memory) with Attention to process longer statements while preserving performance; the model has the ability to remember more features from the input statement and in both directions (reading from left to right and right to left) while learning which terms to pay attention to for proper sentiment classification. An initial implementation for the model was done and tested on the IMDB data set. The First implementations showed a train loss of around 0.04 and test loss of approximately 0.4. It also achieved accuracy of 57%.