COMP 7150/8150 Data Science
FALL 2016

INSTRUCTOR: Dr. Deepak Venugopal

Sentiment Analysis of Presidential Debate 2016

Diem-Trang Pham
Quang Tran

The 2016 election is the most mobile and most social election in American history. Kicking and screaming, the presidential election has caterwauled online, from the primaries to the general election. The first presidential debate of the campaign is nigh, polls are tightening, and voters are listening. In this project, we investigate the sentiments from candidates based on words, sentences, time responses, and other possible factors. We also introduce an automated tool for a process input in csv format so as to get a proper data format.

Introduction

Problem Definition and Algorithm

2.1 Task Definition

Sentiment analysis is extremely useful in social media monitoring as it allows us to gain an overview of the wider public opinion behind certain topics. Sentiment Analysis recently becomes a good method to learn about other people's insight, since the true sentiment of people is not something we can measure by asking. In the presidential debate, the moderator will open each segment with a question, after which each candidate will have two minutes to respond. Candidates will then have an opportunity to respond to each other. The moderator will use the balance of the time in the segment for a deeper discussion of the topic.

In this project, we analyze sentiment from presidential debate since we want to know how candidate react to the question and other candidate's response in a short given time. Input is a raw debate script, which is

provided after every debate. Raw input is processed to organize the data by each speaker. Processed input has following features:

- Line: line of text in the raw input.
- Speaker: name of speaker.
- Text: content.
- Date: date of debate.

Outputs of this project are visualization of several different text analysis.

2.2 Algorithms

In this project, we use several algorithms and techniques to do sentiment analysis.

- Natural Language Processing techniques to process text (remove stop words, get word tokenization, word frequencies, vectorize text,...).
- Logistics Regression to predict sentiment in sentences.
- Word cloud: an image composed of words used in a particular text or subject, in which the size of each word indicates its frequency or importance.
- Other comparisons.

Experimental Evaluation

Methodology

In order to count characters, words, sentences, we used RegexpTokenizer.

```
tokenizer = RegexpTokenizer(r'\w+')
clinton_text = '. '.join(clinton_df['Text'])
trump_text = '. '.join(trump_df['Text'])
clinton_words = tokenizer.tokenize(clinton_text)
trump_words = tokenizer.tokenize(trump_text)
clin_sent_lengths =[len(tokenizer.tokenize(sentence)) for sentence in clinton_sentences]
tr_sent_lengths =[len(tokenizer.tokenize(sentence)) for sentence in trump_sentences]
```

To investigate further to the debate visualization, we like to know candidate is the most given to loquaciousness.

Use this function to remove punctuations and return a list with separated words.

```
def getWords(text):
        return re.compile('\w+').findall(text)
```

Since time between people in the debate cannot be measured precisely, we used word count as a proxy for time.

df["Length"] = df.Text.map(getWords).map(len)

For Logistic Regression, we use a publicly uploaded data (https://dl.dropboxusercontent.com/u/8082731/datasets/UMICH-SI650/training.txt) to train the classifier. Predicted results will rely on how much similar the topic of content in training data with the input text.

Results

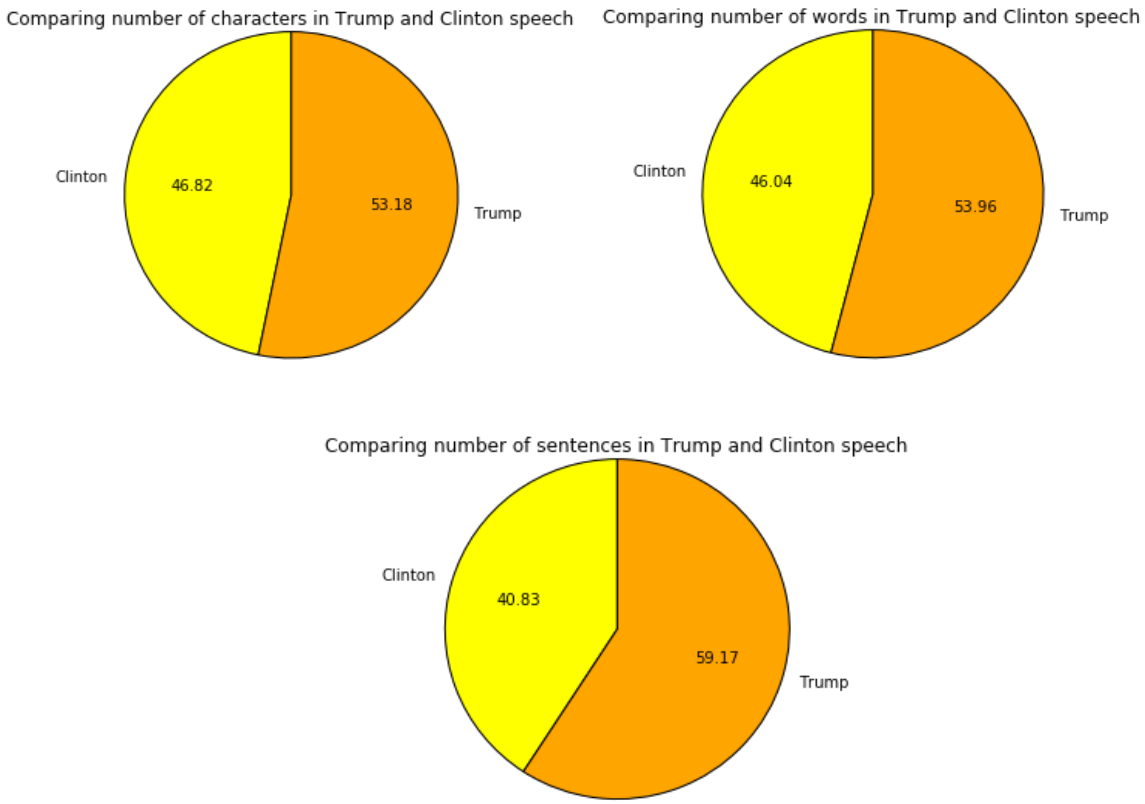

Figure 1. Comparison of number of characters, words and sentences from two candidates.

In this debate, regardless of their sentiment Trump likely made longer and more complex sentences. Since we can not measure exactly speak time of each candidate, we assume two candidate speak at the same speed, then we counted the number of words in each sentence and considered as a response length.

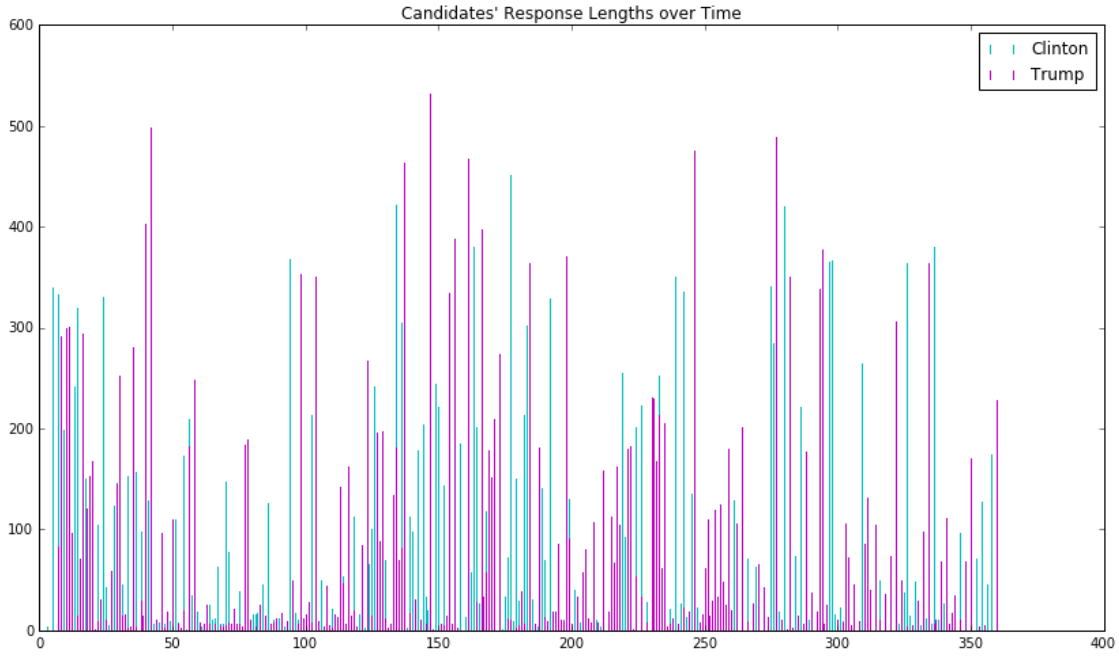

Figure 2.

Utterances are likely expressing words that candidates may consider to value them the most in their speech. After removing all stopwords, all remaining words are used as utterances and we count for their frequencies. So in these figures, Clinton values “people”, “think”, and had used the word “well” as a stop-word for most of her speech. On the other hand, Trump wanted to be “going”, as well as focussed on “people”, and “country”.

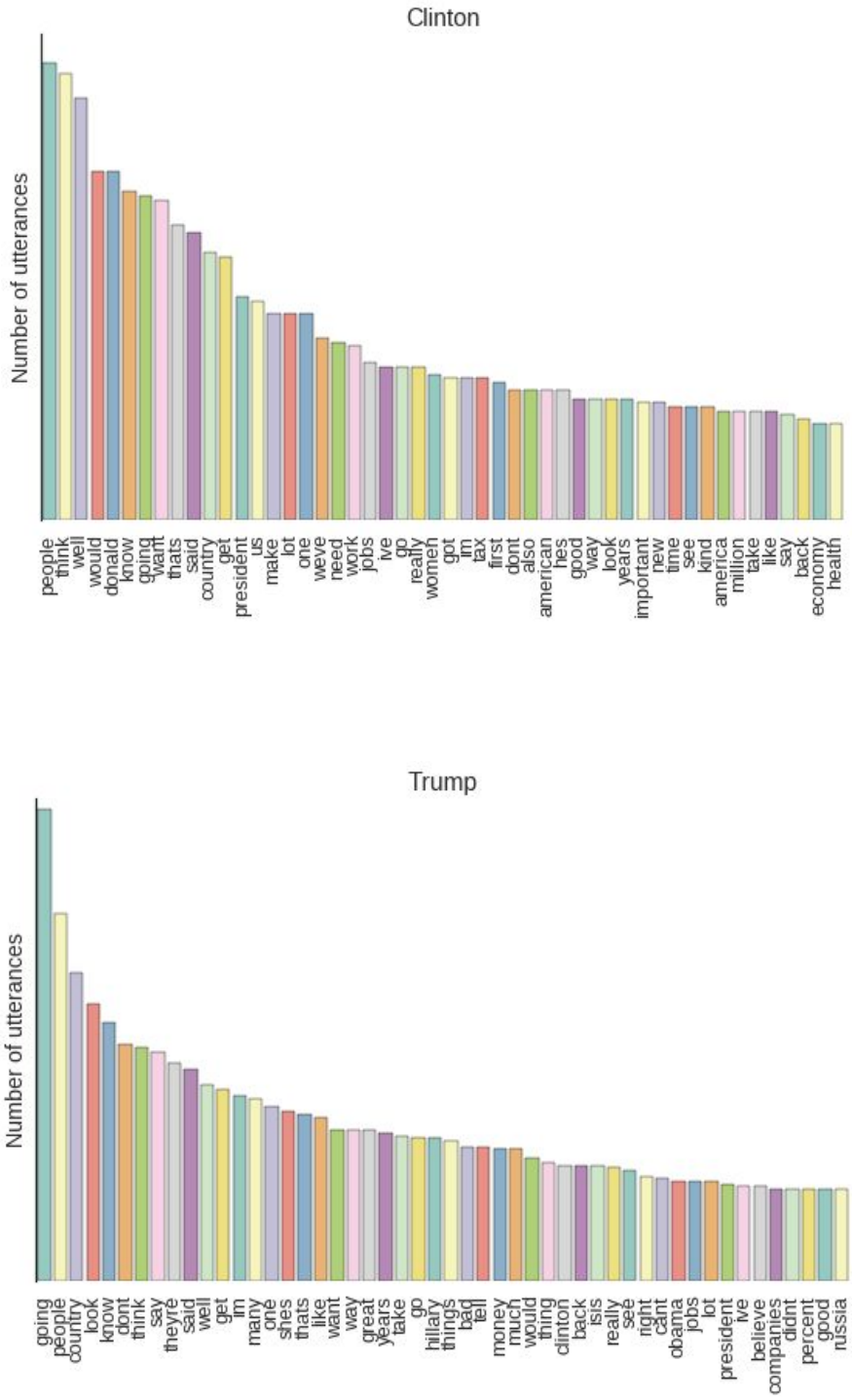

Figure 3. Top 50 utterances used by two candidates.

Figure 4. Percentage of sentiment words used by two candidates.

Clinton used more sentiment words, but Trump has more sentiment sentences overall. It could be because of the training data values more words or sentences in business.

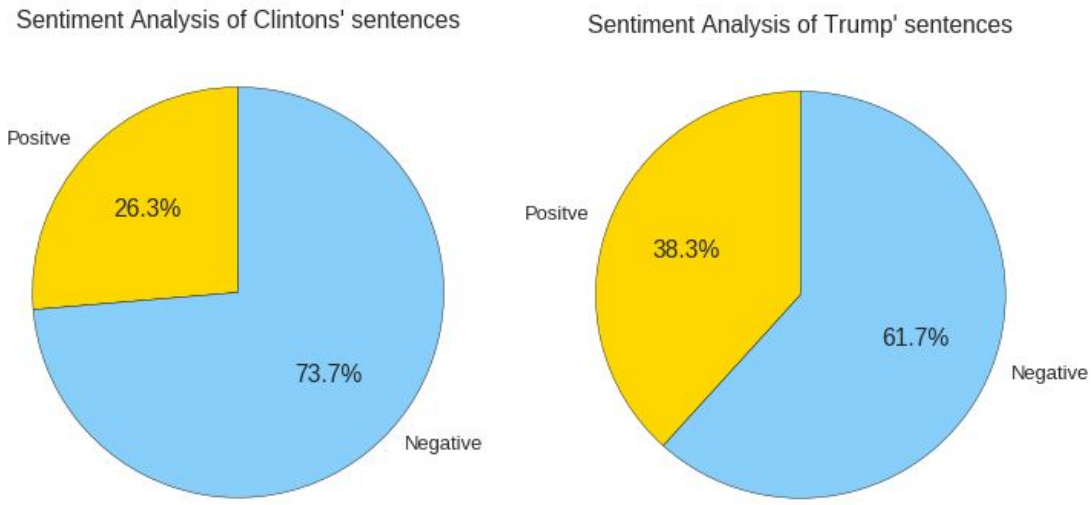

Figure 5. Percentage of sentiment sentences used by two candidates.

Figure 6. Word cloud for two candidates.

From the word cloud, Trump's topic is more extreme while Clinton's topic is more general. Trump's top 10 most used frequent words are mostly related to business, such as "job", "tax", "money", "deal", "company". On the other hands, Clinton mentioned variable topics, such as "work", "woman", "American", "job", "tax", "right", "state".

Conclusion

Sentiment analysis is a powerful tool which can be applied in different area, such as analyzing politicians' thoughts, listening to customers' feedback in business, or making decisions about products and advertising. A person can also influence the conversation if he or she knows which way it is trending, nudging the view to make it positive. However, automated sentiment analysis will never be as accurate as human analysis, because it doesn't account for the subtleties of sarcasm or body language. This project can be extend by analyzing sentiment phrases or combining with video/image sentiment analysis.