SoftDes MP1 Writeup

Ziyu Wang

September 27, 2015

1 Project Overview

My goal for this project is to harvest tweets from Twitter about each presidential candidate that is running in the 2016 election. After I have a sufficient amount of tweets, I would do sentiment analysis and find out the average polarity and subjectivity of tweets about each candidate. I can then identify the candidate with the most positive public impression and the candidate with the most negative public impression. I think this would indicate the public opinion and interest in the candidates and may predict the results of the final election.

My general approach is to use Pattern and Twitter API to collect data and store them in plain text files and Pattern's sentiment analysis to calculate the polarity and subjectivity.

2 Implementation

Text_mining.py first creates a list with all the candidates' names and then reverses it to generate a list with the candidates' names in the opposite order(this is so that later the for loop can traverse the list backwards if reaches Twitter API limit while using the normal-ordered list so we collect data for all the candidates). Then it runs a for loop for each candidate in the list and creates a plain text file under his/her name for storing data. Pattern and Twitter API then collect tweets about the candidate from the Twitter stream for the past 30 seconds and then store that data in the plain text file for later analysis.

Text_analysis.py then runs a for loop for each candidate in the list and creates three lists for storing the entire sentiment analysis, polarity, and subjectivity. It then runs sentiment analysis on each tweet and store the data in corresponding lists. It then finds the average polarity and subjectivity by dividing the sum of the list with the length of the list.

If someone were to run this code, they should run text_mining.py first to collect the necessary data and then run text_analysis.py to get the analysis data.

I used lists because they are mutable and therefore easy to access and manipulate (index, sum, length, etc).

3 Output and Results

Here is an example of the output from text_mining.py.

```
HillaryClinton
RT @finneyk: Questions on @HillaryClinton's emails on this morning's #MTP?
Asked and answered. Time to move on.
So @HIllaryClinton requested a 'private elevator' when living in Spielberg's
luxury Trump Tower Apt.! #HIllary2016 #Wow But she was denied.
RT @TheRightWingM: Trump tonight says: He wants to tax the rich &amp;amp; provide
government run health care. In other words, he's running with @HillaryClinton's
plan
.@HillaryClinton but what about the money you're taking from the plantation
owners and slave catchers? .@POTUS did they give you money too?
I also add that he wants to allow ISIS to overrun Syria &amp;amp; a trade war with
China &amp;amp; Mexico https://t.co/RO2CXriM6L
DonaldTrump
RT @yofun0: @realDonaldTrump If your so good at negotiating, how come you
couldn't get @DonaldTrump
Donald Trump on 60 minutes full interview (09 - 27 - 2015)
#Trump2016 #DonaldTrump #Trump60minutes http://t.co/TCAxwLEqmn
```

And here is an example of the output from text_analysis.py.

```
Number of tweets about HillaryClinton: 307
HillaryClinton's average polarity: 0.0733076239947
HillaryClinton's average subjectivity: 0.257115595226
Number of tweets about DonaldTrump: 256
DonaldTrump's average polarity: 0.0790201822917
DonaldTrump's average subjectivity: 0.319986979167
Number of tweets about BernieSanders: 271
BernieSanders's average polarity: 0.0792178239964
BernieSanders's average subjectivity: 0.153370233702
Number of tweets about BenCarson: 106
BenCarson's average polarity: -0.0398748689727
BenCarson's average subjectivity: 0.206905136268
Number of tweets about JebBush: 198
JebBush's average polarity: -0.0113509349447
JebBush's average subjectivity: 0.154592352092
Number of tweets about TedCruz: 204
TedCruz's average polarity: 0.0437648544266
TedCruz's average subjectivity: 0.116109625668
Number of tweets about MarcoRubio: 128
MarcoRubio's average polarity: 0.0106305803571
MarcoRubio's average subjectivity: 0.128250558036
Number of tweets about MikeHuckabee: 3
MikeHuckabee's average polarity: 0.416666666667
MikeHuckabee's average subjectivity: 0.583333333333
Number of tweets about RandPaul: 112
RandPaul's average polarity: 0.0244419642857
RandPaul's average subjectivity: 0.154761904762
Number of tweets about CarlyFiorina: 49
CarlyFiorina's average polarity: -0.0254616132167
CarlyFiorina's average subjectivity: 0.392371234208
There is no tweets about ScottWalker collected
Number of tweets about JohnKasich: 27
JohnKasich's average polarity: 0.0797558922559
JohnKasich's average subjectivity: 0.389917695473
```

```
There is no tweets about MartinO'Malley collected
Number of tweets about ChrisChristie: 28
ChrisChristie's average polarity: 0.155505952381
ChrisChristie's average subjectivity: 0.329761904762
There is no tweets about JimWebb collected
Number of tweets about RickSantorum: 25
RickSantorum's average polarity: -0.074
RickSantorum's average subjectivity: 0.22
Number of tweets about BobbyJindal: 12
BobbyJindal's average polarity: 0.6
BobbyJindal's average subjectivity: 0.7875
There is no tweets about LincolnChafee collected
There is no tweets about LindseyGraham collected
There is no tweets about GeorgePataki collected
There is no tweets about JimGilmore collected
There is no tweets about JillStein collected
```

As you can see from the results, Rick Santorum has the most positive public impression (polarity: 0.6) but relatively low public interest (12 tweets) while Bernie Sanders has the second most positive public impression (polarity: 0.079) but relatively high public interest (271 tweets). Rick Santorum has the most negative public impression (-0.074) but relatively low public interest.

4 Reflection

In general, the whole project went pretty well. The text mining part was a bit tricky having to deal with Twitter API, its access limit, and the specific Python commands for interacting with .txt files. I could have combined the two scripts but I wanted to be able to collect data multiple times without running analysis on them every time. I could have compared data from different weekdays and see if that would have affected people's view or compared data before and after some big news and see how people's opinion were affected. This project was appropriately scoped for the time given and I learned a lot doing the project. Let's see if the results will confirm my prediction when the election comes around.