Analysis of public sentiment on Twitter versus debate performance of top GOP candidates.

Dealing with Data Final Project- TEAM #2

Team Members: Ankur Chaudhary , Christina Wei, Janul Hernandez, and Peter Meijer

5/9/2016

NYU STERN

Executive Summary

As a GOP candidate has Donald Trump seen more success from negative rhetoric than his running mates?

Project Process

- Which sources of data did we look at?
- Difficulties in acquiring data.
- How did we use the data.

What do the debates say?

A focused look at debate sentiment analysis per candidate through the last 3 debates of the GOP season.

Public Sentiment?

How is Trump trending on Twitter compared to his running mates?

Conclusion

- What does the analysis and data tell us
- What could we have done different?

Agenda

- **Project Process**
- What Do the Debates Say?
- Public Sentiment
- Conclusion

Project Process

Data Source
- Washington Post
- Twitter

Collection
- BeautifulSoup
- Tweepy

Analysis
- Alchemy API
- Text Sentiment

Manipulation
- mySQL

Visualization
- Matplotlib
- Tableau

Difficulties

```
print openFile(debates[0])
```

```
<!DOCTYPE html>
<html class="blog layout_article" itemscope="" itemtype="http://schema.org/NewsArticle" lang="en"> <head> <meta content="IE=ed
ge,chrome=1" http-equiv="X-UA-Compatible"/> <meta charset="utf-8"/> <meta content="width=device-width, initial-scale=1.0, user
-scalable=yes, minimum-scale=0.5, maximum-scale=2.0" id="viewport" name="viewport"/> <meta content="unsafe-url" name="referre
r"/> <meta content="Donald Trump, Ted Cruz, Marco Rubio, Republican debate, John Kasich, Ben Carson" name="keywords"/> <meta c
ontent="Read the transcript of the last Republican debate before Super Tuesday, annotated with insight and analysis from our r
eporters." itemprop="description" name="description"/> <meta content="Donald Trump, Ted Cruz, Marco Rubio, Republican debate,
John Kasich, Ben Carson" name="news_keywords"/> <meta name="twitter:site" value="@WashingtonPost"/> <meta content="summary_lar
ge_image" name="twitter:card"/> <meta content="Read the transcript of the last Republican debate before Super Tuesday, annotat
ed with insight and analysis from our reporters." property="og:description"/> <meta content="article" property="og:type"/> <me
ta content="Washington Post" property="og:site_name"/> <meta content="True" itemprop="mainEntityOfPage"/> <meta content="http
s://www.facebook.com/washingtonpost" property="article:publisher"/> <meta content="1513210492" property="fb:admins"/> <meta co
ntent="41245586762" property="fb:app_id"/> <meta content="1244584375" property="fb:admins"/> <meta content="4403963" property
="fb:admins"/> <meta content="2724956" property="fb:admins"/> <meta content="500835072" property="fb:admins"/> <meta content
="app-id=938922398, app-argument=https://www.washingtonpost.com/news/the-fix/wp/2016/02/25/the-cnntelemundo-republican-debate-
transcript-annotated/" name="apple-itunes-app"/> <link href="https://www.washingtonpost.com/pb/resources/json/manifest.json" r
el="manifest"/> <link href="https://www.washingtonpost.com/news/the-fix/wp/2016/02/25/the-cnntelemundo-republican-debate-trans
cript-annotated/" itemprop="url" rel="canonical"/> <meta content="https://www.washingtonpost.com/news/the-fix/wp/2016/02/25/th
e-cnntelemundo-republican-debate-transcript-annotated/" property="og:url"/> <meta content="http://www.washingtonpost.com/blog
s/the-fix/files/2016/02/APTOPIX_GOP_2016_Debate-0e04c.jpg" itemprop="image" property="og:image"/> <meta content="The CNN-Telem
```

Washington Post

- Parsing html
- Determining who said what
- Removing punctuation

Twitter

Tweepy

- 429 Overload
- Geocode & User Location – user enabled
- 3200 tweets max per unique query

Topsy

- Shut down – 10 days max twitter history

Sentiment Analysis

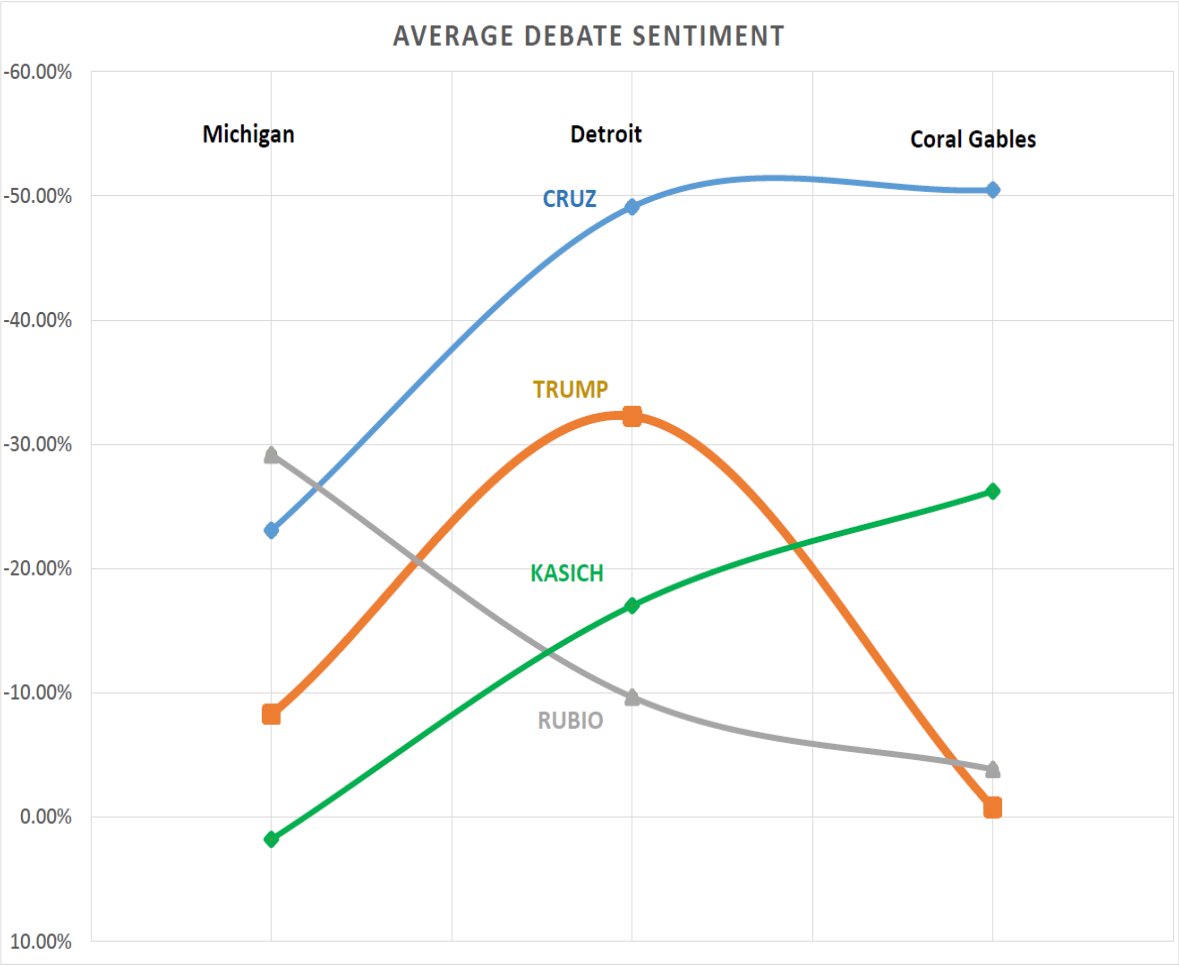
- Throttle error

Traceback (most recent call last): File "extract_test.py", line 43, in response=br.open(v) File ""/usr/local/lib/python2.7/dist-packages/mechanize/_mechanize.py" line 203, in open return self._mech_open(url, data, timeout=timeout) File "/usrlocal/lib/python2.7/dist-packages/mechanize/_mechanize.py", line 255, in _mech_open raise response mechanize._response.httperror_seek_wrapper: HTTP Error 429: Unknown Response Code

Agenda

- Project Process
- **What Do the Debates Say?**
- Public Sentiment
- Conclusion

What do the debates say?

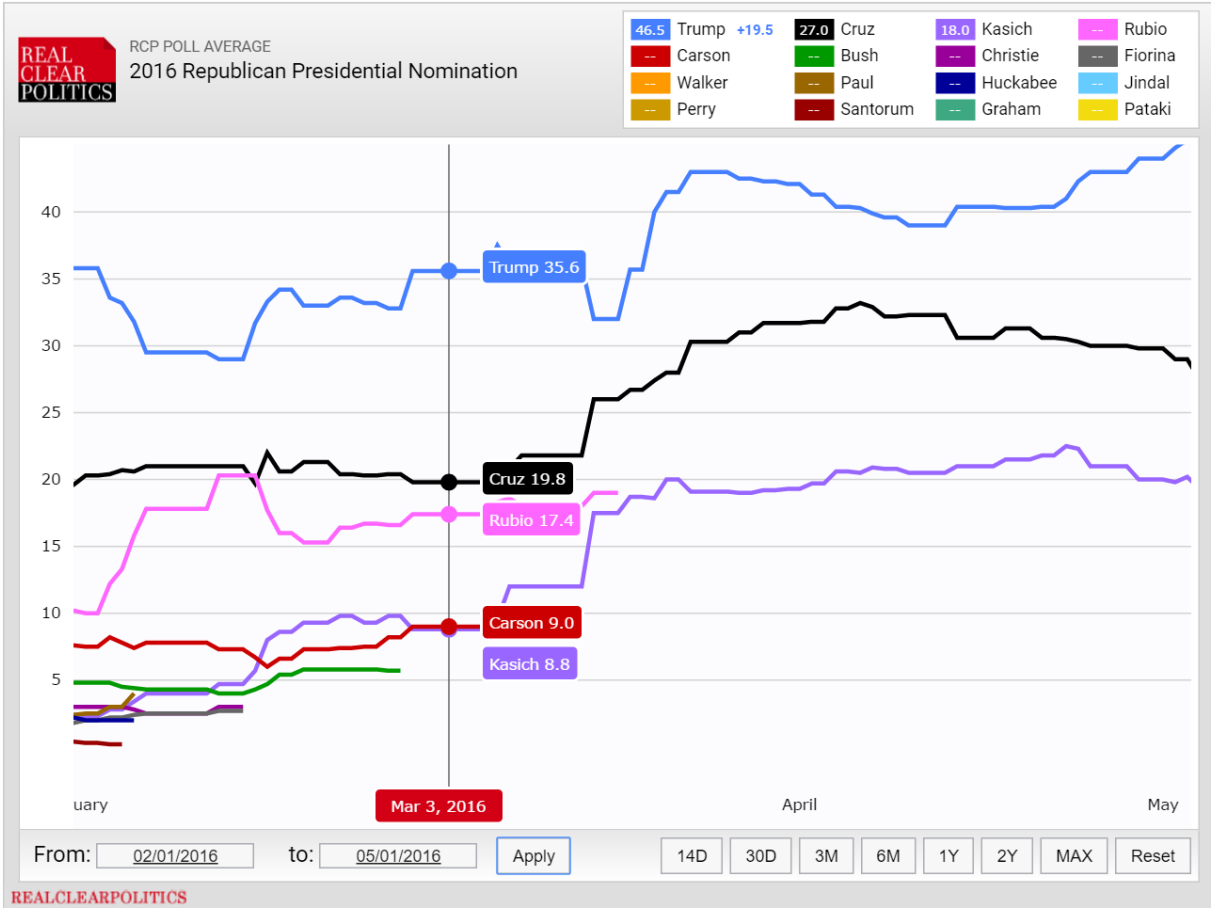


$$Sensitivity = \frac{\sum_{i \to I}(i_{pos} - i_{neg})}{\sum i}$$

$$where\ i = relevance$$

- Cruz and Kasich get more negative as they get desperate.
- Trump has an outlier negative performance in Detroit
- Rubio continues to be neutral

What happened in Detroit?

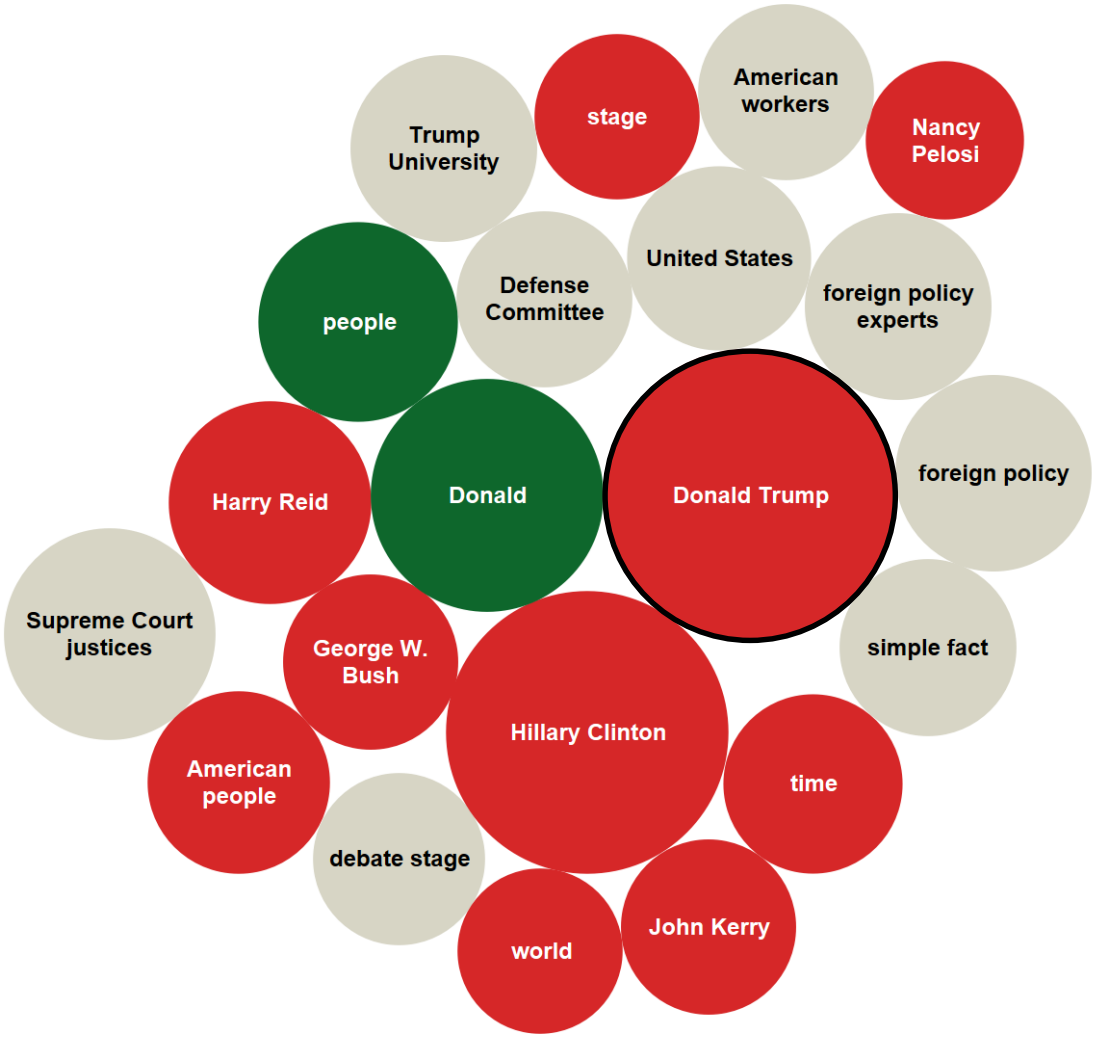


NEWS HEADLINES

- The Washington Post
  "Trump was attention of attacks at GOP debate"
- The New York Times
  "Taking On Donald Trump, at Their Own Peril"
- Rolling Stone
  "Watch every time FOX News attacked Trump at GOP debate."

NYU STERN

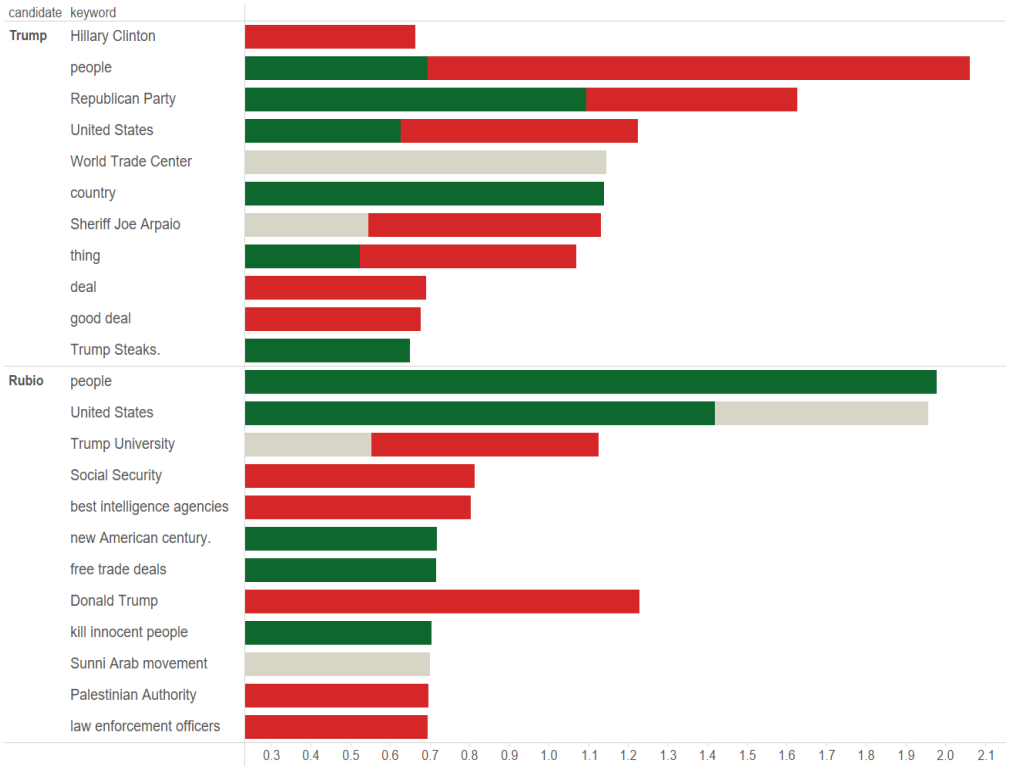
What happened in Detroit?

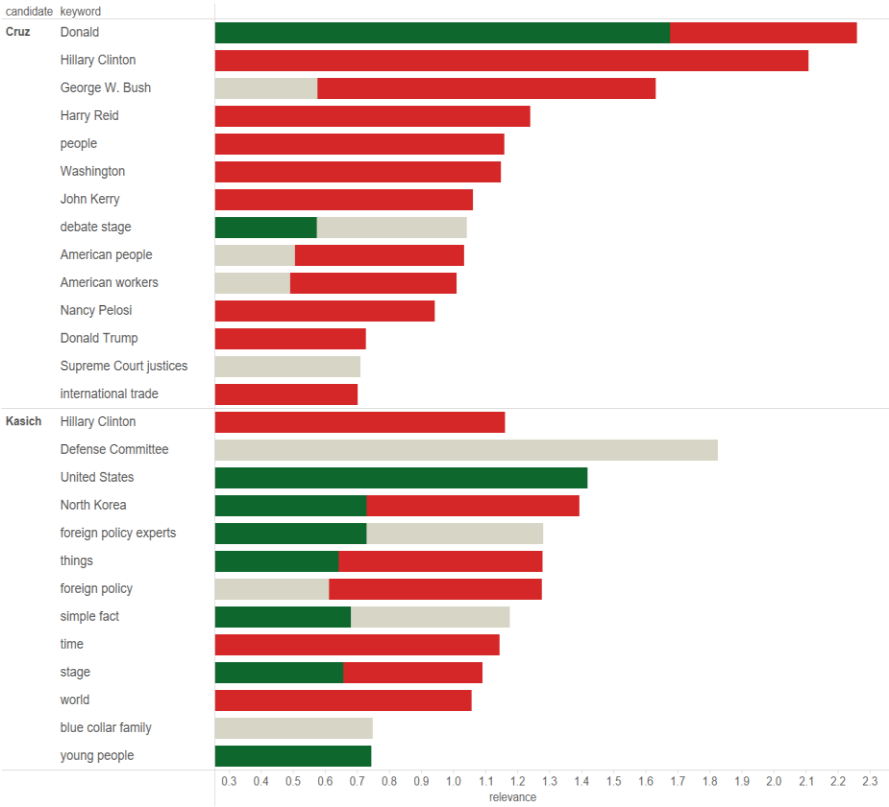


Trump proving he doesn't have small hands

NYU STERN

Key Topics Through 3 Debates

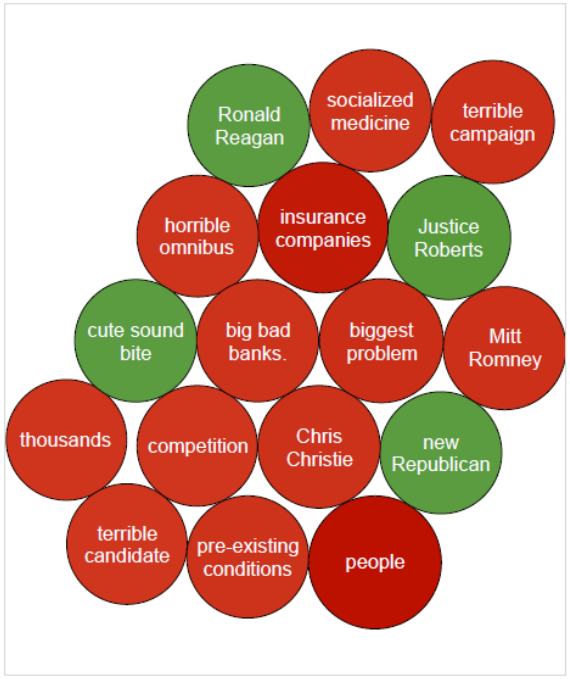




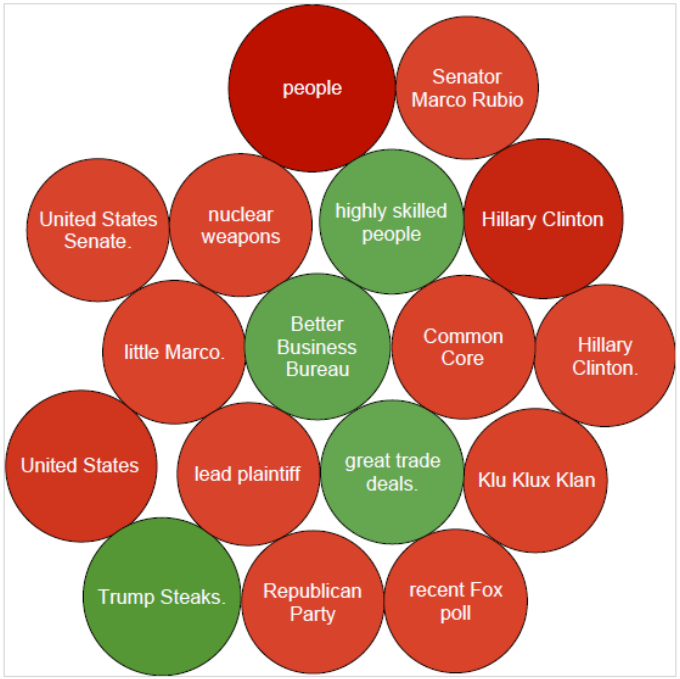
NYU STERN

Trump Main Focus

Debate1



Debate2



Debate3

- Common theme of the people not being happy.
- Denouncing government veterans and current GOP establishment.

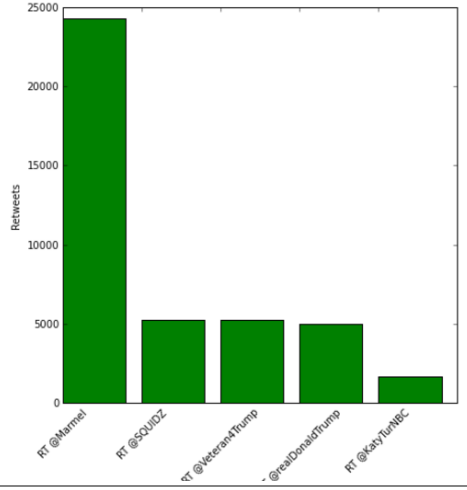
Agenda

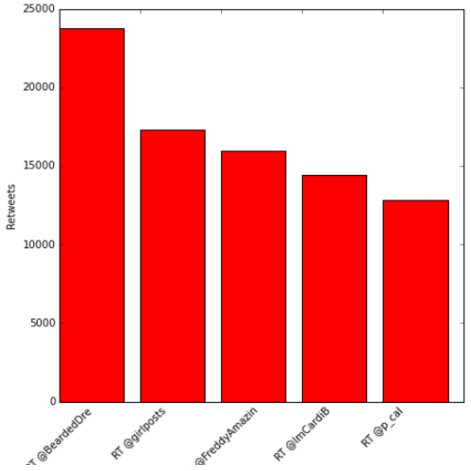
- Project Process
- What Do the Debates Say?
- **Public Sentiment**
- Conclusion

Trump: Top 5 Negative and Positive Tweets

	POSITIVE
1	Whoever made this Trump v. Manson. http://t.co/ztUx36aY1c
2	Mexicans and Black People teaming up against Donald Trump BOY LIFE BEAUTIFUL RN MY NIGGA
3	I'm a Veteran. I was born in Mexico
4	Thank you Wilkes-Barre #MakeAmericaGreatAgain #Trump2016 https://t.co/FJbNlidwJF
5	Wilkes Barre

	NEGATIVE
1	I pray and I pray trump doesn't get elected VOTE…
2	Me: green is like the ugliest color ever Donald trump: I hate the color green Me: https://t.co/ryJcjKm8B1
3	I'M SCREAMING THEY DROVE PAST A TRUMP RALLY AND PLAYED THIS https://t.co/gZAnYw1sXs
4	Donald trump gona deport all the foreign bitches that yall love fucking
5	Donald Trump rally vs Bernie Sanders rally



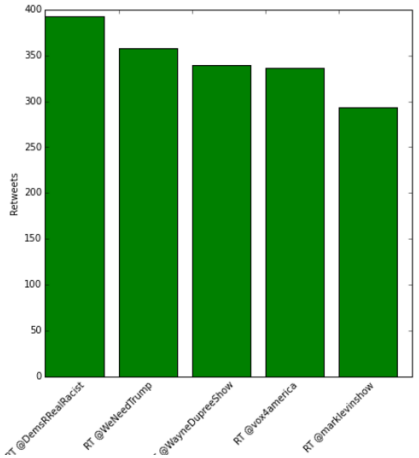


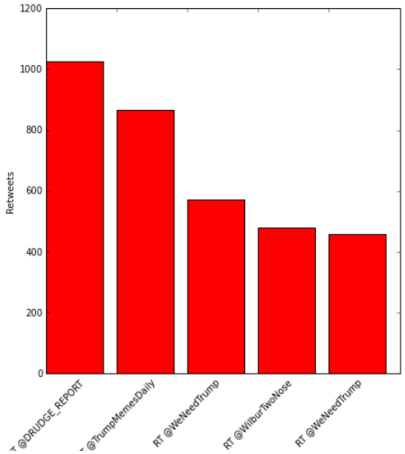
NYU STERN

Cruz: Top 5 Negative and Positive Tweets

	POSITIVE
1	Ted Cruz understands that step number one is defeating that nasty Washington cartel is owing your electoral successes…
2	Cruz
3	Now do you believe me that the Cruz campaign is struggling #TrumpTrain #trump…
4	On behalf of Trump supporters
5	Sellout? Nah

	NEGATIVE
1	CRUZ CONFRONTED: HOW CAN YOU HAVE DELEGATES WITHOUT A VOTE? https://t.co/iUejLichH
2	Cruz and Kasich should just drop out already #Trump2016 #VoteTrump #InPrimary #MakeAmericaGreatAgain
3	Ithe people aren't happy with the dirty politics Cru is playing. It's over
4	Cruz on CNN just said
5	RETWEET if you think Cruz and Kasich's attempt to subvert the popular vote is pathetic! #Trump2016



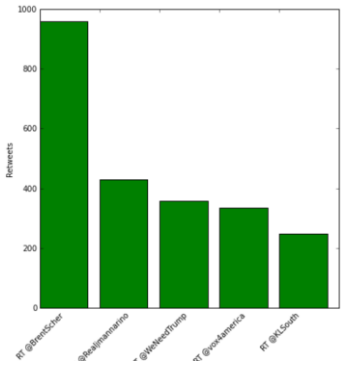


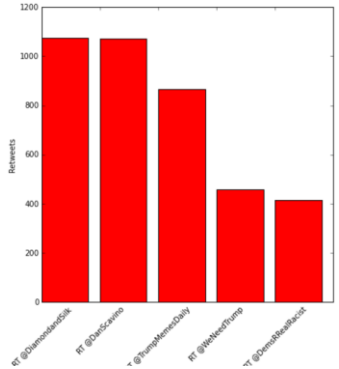
NYU STERN

Kasich: Top 5 Negative and Positive Tweets

	POSITIVE
1	Am I crazy Kasic
2	RETWEET THIS POLL SO WE CAN GET A BIG PICTURE IDEA! Cruz and Kasich teaming up makes me…
3	Cruz Kasic
4	On behalf of Trump supporters Kasich
5	#Cruz #Kasich are publicly confessing they're nothing more than spoilers on behalf of the Establishment – as #Trump expands hi…

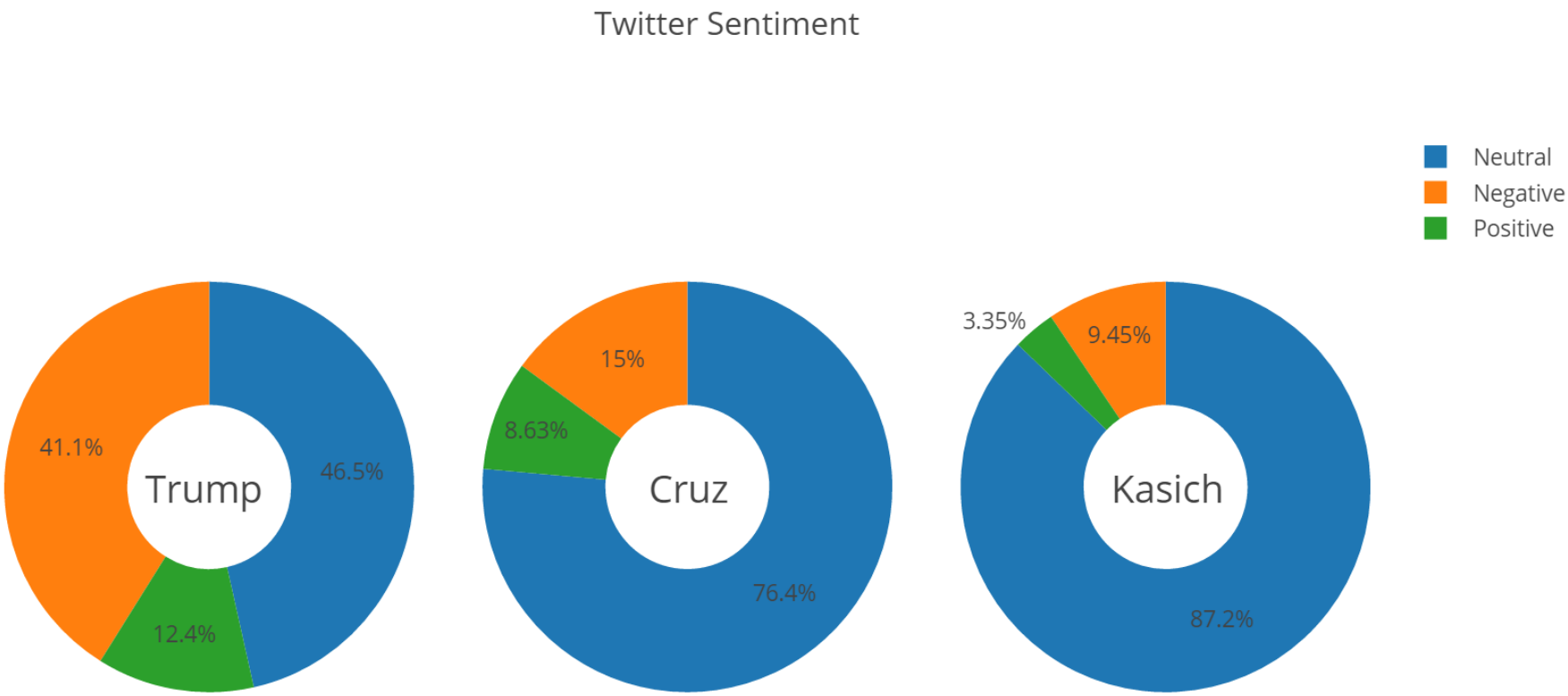
	NEGATIVE
1	Lyin' Ted & Lunatic Kasich it will be a cold day inside of a Volcano before @DiamondandSilk vote for either of you.
2	Should say: Follow taking points in Kasich deal. WE ARE LOSING & WE ARE DESPERATE! Don't let them see you sweat!
3	Cruz and Kasich shoul djust drop out drop out already #Trump2016 #VoteTrump #INPrimary #MakeAmericaGreatAgain https://…
4	RETWEET if you think Cruz and Kasich's attempt to subvert the popular vote is pathetic! #Trump2016
5	Attention Cruz and Kasich supporters: read your e-mail CAREFULLY so you don't get confused about who they say you have…





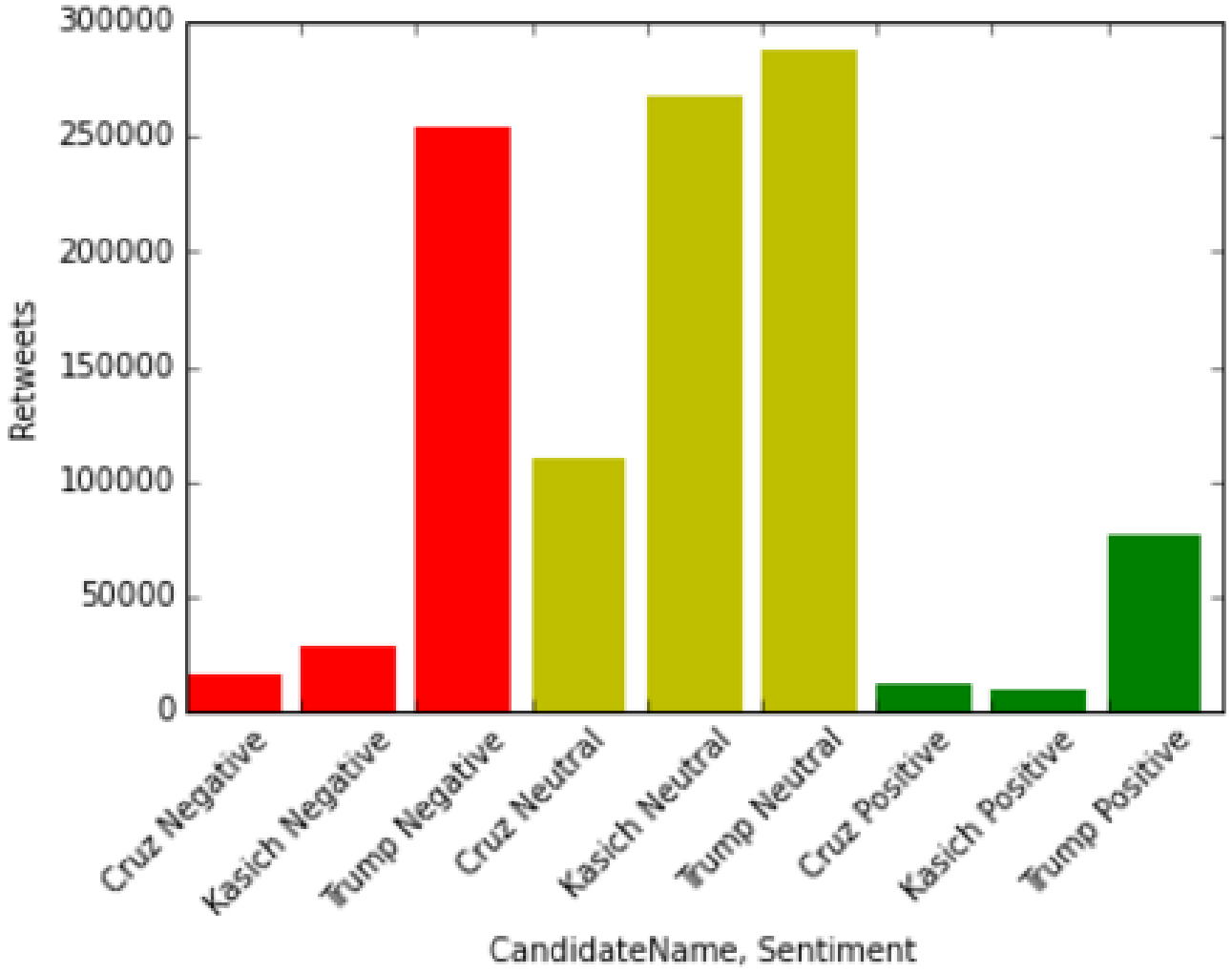
NYU STERN

Trump Wins the Popularity Vote

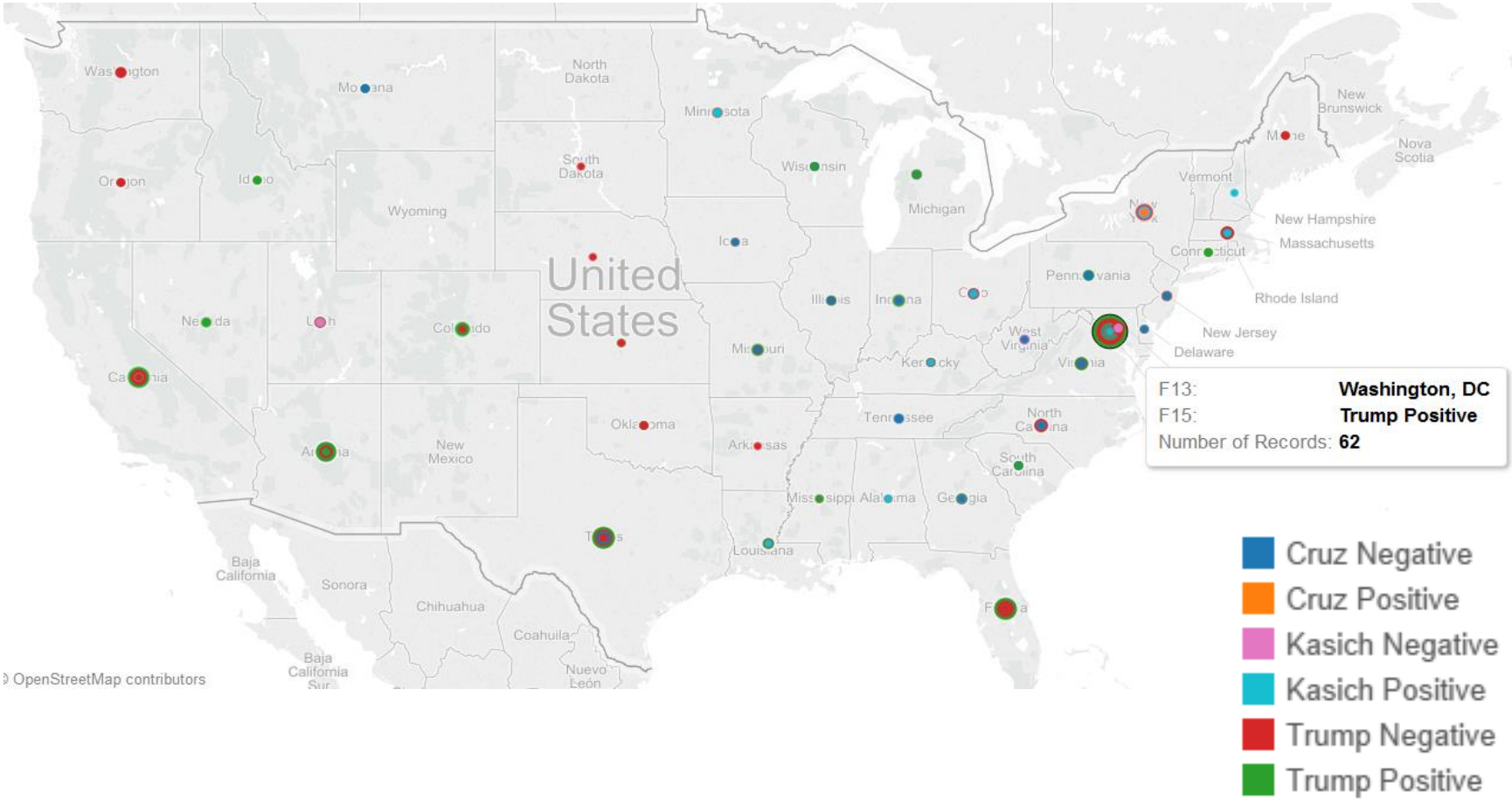


Although all candidates' tweets have similar sentiment profiles, Trump wins due to the sheer volume of tweets about him

Candidate Sentiment by Volume

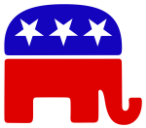


Candidate Tweets by Map



Agenda

- Project Process
- What Do the Debates Say?
- Public Sentiment
- **Conclusion**



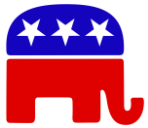
Conclusion

Our hypothesis erred on two key aspects

–Assumption: Trump most negative debate sentiment
–Assumption: Trump most positive social media support

Ultimately, Trump's debate sentiment *improved* over the course of the debates while his primary challenges Cruz and Kasich grew more negative

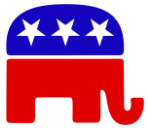
Trump generated the most social media, but also accounted for the **most negative** social media due to heightened general scrutiny of controversial comments



Additional Questions

Our approach yielded additional research opportunities:

1. Using Twitter sentiment analysis and location data of tweets, examine correlation between change in tweet sentiment pre/post-debate and whether debate sentiment had an impact on polling data relative to state primary results
2. For the debates between Donald Trump and the democratic candidate, apply a similar analysis over several debates to chart the impact of debate sentiment for a more constrained data sample

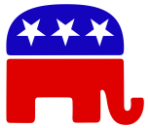


Thank You

Thank you for your attention. We welcome any comments or questions on our project.

Sincerely,

Janul, Ankur, Christina, & Peter



Appendix

2 Defining global functions.

```
In [ ]: def openFile(url):
            #opening files
            file0 = BeautifulSoup(urllib.urlopen(url).read(),"lxml")
            return file0

        def cleanFile(file1):
            #function extracts text lines from html
            textLines = []

            paragraph = file1.find_all("p")
            for lines in paragraph:
                textLines.append(lines.get_text())

            return textLines

        def lines2words(file1):
            #function takes a list of strings and returns it as a string of words
            return_string = ''

            for line in file1:
                return_string += line

            words = return_string.split()

            return words
```

```
def getCandidate(candidate,file2):
    #function gets cadidate lines
    i = 0
    candidateLines = []

    for line in file2:
        if candidate in line:
            candidateLines.append(line)
            i += 1
        elif candidate in file2[i] :
            candidateLines.append(line)
            i += 1
        else:
            i += 1

    return candidateLines

print openFile(debates[0])
```

Appendix

3 Function to compile all keywords due to API time out limit

```
In [ ]: def keywordAnalysis (textFile):
        #Calculating how many passes to send the API through according to dictionaryt length.
            multiple= 0
            multiple= len(textFile)/50
            remainder= len(textFile)%50
            if remainder > 0:
                multiple += 1

            temp=[]


            url = "http://access.alchemyapi.com/calls/text/TextGetRankedKeywords"
            api_key = '1f62c149d1b092011db73353df8215f6a5d3eb9e'
            headers = {"Accept": "application/json"}

            i=0
            while i != multiple:
                text = textFile[i*50:i*50+50]

                parameters = {
                    'outputMode': 'json',
                    'apikey' : api_key,
                    'maxRetrieve' : 200,
                    'sentiment': 1,
                    'text': text}

                resp = requests.post(url, params=parameters, headers=headers)
                temp.append(json.loads(resp.text))
                i += 1
```

Appendix

4 Final Function.

You choose a candidate name and provide a source of text. The function parses the data and returns back keywords according to relevace.

```
In [ ]: def getCandidateKeywords(candidateName,URL):

            File1 = openFile(URL)
            textFile = cleanFile(File1)

            candidateLines= getCandidate(candidateName, textFile)

            word_counts = collections.Counter(lines2words(candidateLines))

            #return keywordAnalysis(candidateLines)
            return word_counts

            #for word, count in word_counts.most_common():
                #print word, count
```

Appendix

8 Function to send sentiment analysis to SQL

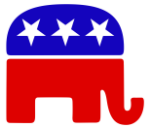
```
In [ ]: def send2SQL (candidateName, DebateSent, DebateDic):
            cursor = con.cursor()
```

```
query_template = 'INSERT INTO Final_Project.'+ DebateSent + '(candidate, keyword, relevance

for line in DebateDic[candidateName]:

    i = 0
    while i != len(line['keywords']):
        candidate = candidateName
        keyword = line['keywords'][i]['text']
        relevance= line['keywords'][i]['relevance']
        sentiment= line['keywords'][i]['sentiment']['type']
        print "Inserting ", candidate, keyword

        query_parameters = (candidate, keyword, relevance, sentiment)
        cursor.execute(query_template, query_parameters)
        con.commit()
        i+=1


cursor.close()
```

Appendix

Inserting data into CSV via Tweepy and running Sentiment classifier over it

```
auth = tweepy.OAuthHandler(CONSUMER_KEY, CONSUMER_SECRET)
auth.set_access_token(ACCESS_KEY, ACCESS_SECRET)

#Trump API

api = tweepy.API(auth)
# Append data to CSV
CSVF = open('/home/ubuntu/data/Trump.csv', 'a')
CSV Writer
CSVW = csv.writer(CSVF)
#most recent tweet id for iteration
lis = [724910184376750080]
x=0
total_retweets=0

#print the_page

for i in range(0, 16):
    time.sleep(40)
    for id,tweet in enumerate (tweepy.Cursor(api.search,
                    q='Trump',
                    since="2016-04-26",
                    until="2016-05-03",
                    lang="en",
                    max_id=lis[-1]
                    ,include_retweets=False
                    ).items(200)):
        #Write a row to the csv file/ I use encode utf-8
        data = urllib.urlencode({"text":tweet.text.encode('utf-8')})
        u = urllib.urlopen("http://text-processing.com/api/sentiment/", data)
        the_page = u.read()

        lis.append(tweet.id)
        total_retweets+=tweet.retweet_count+ 1
        CSVW.writerow(['Trump',tweet.geo,tweet.place,tweet.created_at,tweet.retweet_count, the_page, tweet.text.encode('utf-8')])
        print x,'Trump',tweet.created_at, tweet.text,tweet.retweet_count,the_page
        print "Retweets Till now :",total_retweets
        x+=1
CSVF.close()
```



Appendix

Reading and Transforming CSV data for Trump

In [ ]:
```
!head /home/ubuntu/data/Trump.csv
```

In [ ]:
```
!cut -f1,2,6,7,8,9,10,11 -d',' /home/ubuntu/data/Trump.csv > /home/ubuntu/data/Trump_reduced.csv
```

In [ ]:
```
!head /home/ubuntu/data/Trump_reduced.csv
```

Loading candidate data from CSVs into the database table

In [ ]:
```
!mysql -u root password=dwdstudent2015 --local-infile FinalProject

# It can be done in Terminal as well

cur = connection.cursor()
database = 'FinalProject'
table = 'debate1'
Q2 = '''load data local infile '/home/ubuntu/data/Trump_reduced.csv' into table FinalProject.Republican_debate
fields terminated by ','  enclosed by '"'
lines terminated by '\n'
ignore 1 rows;'''.format(database, table)
```

Appendix

Counting Positive, Negative and Neutral Tweets (with Retweet)

In [ ]:
```
### Count of Tweets + Retweets as Neutral/Positive/Negative for candidates
%%sql select CandidateName, SUM(CASE Sentiment WHEN 'ral"}' THEN (Retweets+1) else 0 END) AS Neutral,
SUM(CASE Sentiment WHEN 'pos"}' THEN (Retweets+1) else 0 END) AS Positive,
SUM(CASE Sentiment WHEN 'neg"}' THEN (Retweets+1) else 0 END) AS Negative
from FinalProject.Republican_debate
where Semantics not like '%Throttled, wait%'
and Candidatename in ('Trump','Cruz','Kasich')
Group by CandidateName
```

```
tr4= %sql select CandidateName,Sentiment,SUM(Retweets+1) as Retweets from FinalProject.Republican_debate  where Sentiment in ('Neutral','Positive','Negative') Group by Sentiment,CandidateName
```