Group Members: Azka Javaid and Caleb Ki
Title: Analysis of the 2016 Presidential Campaign
Purpose:
We are interested in tracking perspectives surrounding the 2016 presidential campaign. Besides the relevance of the topic given the approaching election day, this work is intended to explore the cultural/social framework of the campaign, one liable to change over time by news coverage along with the candidates' continued media and financial investments to maintain their social image. Moreover the campaign has also produced fractures within the parties, especially the Republican candidacy. It would be worthwhile to analyze how the fear and anxiety cultivated by the campaign along with the fissures produced within the political sphere get accentuated or dampened by social media avenues like Twitter and Facebook.

Ideas/Potential End Products:
To analyze changes in sentiments surrounding the campaign, we can use Twitter API to assess emotional valence of tweets, a possible indicator for candidates' popularity. IBM AlchemyAPI also facilitates text analysis (keyword/taxonomy extraction and sentiment analysis). This analysis can lead to predictive/machine learning model, which can be used to assess candidates' popularity/predict the winning candidate on the election day, for example. Since the GDELT data contains emotion analysis, we can use the GDELT election day data to assess our predictions. In tying GDELT with Twitter data, we would analyze the role of social media, networking, and traditional media on influencing election outcomes.

Along the same lines, we can analyze how the presidential campaign influences social media views/sentiments regarding issues like immigration, economy, and defense and analyze how sentiments surrounding these issues evolve after the presidential debates.

We can also assess news source bias in the campaign coverage through the GDELT data (which contains the article tone information) and then use Google Maps API to analyze how the bias varies geographically through various regions and states. Besides GDELT tone variable, news source bias can also be analyzed through text and sentiment analysis (AlchemyAPI) on the news source link. Given that GDELT contains news coverage data from around the world, it may also be worthwhile to assess changes in the presidential campaign around by country regions.

Along the same lines, we can assess the audience breakdown of the candidate's supporters by analyzing Facebook likes (on candidates' profiles) for example, and assess the gender/ethnic makeup of those likes. Much news has also been generated about Facebook's liberal trending bias so we can also use the GDELT data to analyze the accuracy of the trending display and see how closely the trending label actually aligns with the recent world happenings instead of supporting a particular point of view.

Another potential idea would be to analyze how candidate funding affects their popularity. Popularity can be assessed through Twitter API. Google trends package (gtrendsR), which

returns all google searches associated with a term, can also be used to assess candidate popularity.

Data:
Subsets of the GDELT (Global Database of Events, Language, and Tone) is accessible through the GDELT Analysis Service (free cloud-based service). The Database is also accessible through the Google BigQuery platform. Additionally the raw data files for 2015 (>2.5 TB) are accessible. These raw data files contain the events csv file (which contains Date, event actor information and event code). The raw data file also contains a Global Knowledge Graph which contains graphic connections of people, organizations, locations, and emotions. The graph can map predefined data (i.e. map number of people affected by an incident). The Project site also links to a blog which describes the current projects pursued with this dataset accessible at http://blog.gdeltproject.org.

As mentioned above, we plan to use Twitter API to extract campaign related tweets and use AlchemyAPI to perform sentiment analysis. We also plan to use Google Maps API to geographically map sentiments/tweets/candidate popularity. In addition, we would like to incorporate presidential funding data to analyse trends in candidate popularity by the campaign financial investment.

Variables:

EVENTS Table
Variables:
Event Action Attribute:
GlobalEventID (integer): uniquely identifies each event record
Day (integer)/MonthYear(integer)/Year/FractionDate: Identify time of event
NumMentions (integer): total number of mentions of this minute across all source documents during the 15 minute update in which it was first seen.
NumSources (integer): total number of information sources containing one or more mentions of this event during the 15 minute update in which it was first seen.
AvgTone (numeric): average tone of all documents containing one or more mentions of this event during the 15 minute update in which it was first seen. Ranges from -100 (extremely negative) to +100 (extremely positive).

Actor1 and Actor2 Attributes (attributes of two actors involved in event):
ActorCode (String): unique identifier
ActorName (String): Lists name for a political/organization (United Nations), lists country or capital name for geographic reference
ActorCountryCode (String): Identifies country affiliation of the Actor
ActorEthincCode (String): Ethnicity of the actor
ActorReligionCode (String): Religious affiliation of the actor

Event Geography:

ActorGeo_Type (integer): specifies geographic resolution of the match type (filter geographically specific actor)

ActorGeo_Lat and ActorGeo_Long: Latitude and longitude of the landmark

MENTIONS Table (records each mention of the events and tracks the sources the story flows from)

Variables:

MentionType (integer): Identifies the source collection of the document like whether it is web, citation only, CORE, JSTOR, NonTextualSource

MentionSourceName (integer): identifier of the source of the document (like whether the event is BBC Monitored)

SentenceID (integer): sentence within the article where the event was mentioned.

Confidence (integer): percent confidence in the extraction of this event

End Product:

This project can translate in a web app that tracks sentiments/perspectives surrounding the presidential campaign overtime. Additionally, a shiny application and machine learning predictive models (ex. a model that predicts the percentage winning of the candidates by their reputation as measured from twitter sentiment analysis can be tested on the data from election day) will also be potential end products. R package also sounds like an idea further pursuing.