Spring 2019
CMPE 256
Large Scale Analytics

Project Report
Twitter 2012 US Presidential Election Sentiment Analysis

Team Bazinga

Presented to :
Dr. Magdalini Eirinaki

Group Members :
Anisha Vaghela (ID : 011548771)
Anjani Mallampati (ID : 009688627)
Mrunali Sanjay Khandat (ID : 013723177)

1. Introduction

1.1 Motivation

With the rise of social media, users have been able to create communities in which they can share information, opinions, ideas, and other content. Social media has become an integral part of people's lives. People have become more aware of events that occur in the world due to the speed that information can be released with the help of various social media platforms. Twitter, a leading social media platform, has over 300 million active users, including celebrities, politicians, and businesses. Twitter provides a global stage for people to voice their opinion.

Sentiment analysis is "the process of computationally identifying and categorizing opinions expressed in a piece of text, especially in order to determine whether the writer's attitude towards a particular topic, product, etc. is positive, negative, or neutral."

We have chosen the Twitter 2012 USA presidential election dataset because it was a filtered dataset that only included tweets about the election. Second, because this election already took place, we would know the results for comparison.

1.2 Objective

The Twitter 2012 USA presidential election dataset is a compilation of all tweets that discuss the election or the candidates. The aim is to analyze the sentiment of the tweets and determine the percentage of tweets that support Barack Obama and the percentage of tweets that support Mitt Romney. Using the percentages calculated, a comparison will be done using the actual results of the election.

2. System Design & Implementation

2.1 Algorithms

The main goal of our project is to perform sentiment analysis on the tweets of the users that actively participated in voicing their opinion of the presidential election in 2012. Our dataset consists of many different attributes, including the user information, location, hashtags, and much more. Out of all this information, we chose to focus on the text of the tweets, hashtags, and username attributes for finding the sentiment of the tweet. Our approach is to first determine whether the tweet is positive, negative, or neutral. Then, we find the number of positive and negative sentiment towards Obama and Romney, which is used to predict the winner of the election. The dataset that we used for this project does not contain labels, making it hard to measure the accuracy using a traditional method. Thus, we have compared our results with the actual election results.

In order to predict the election results, we have used the following methods:

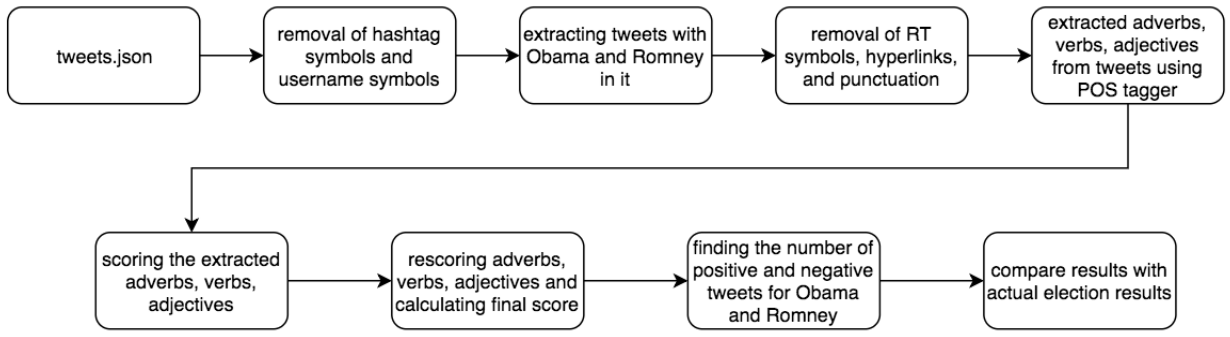
Method 1:

After preprocessing, we get a list of the tweets that contain Obama or Romney. The hyperlinks, RT symbols, and punctuation are then removed from the tweet. Next, we used SentimentIntensityAnalyzer from the nltk.sentiment.vader module of the nltk.sentiment library. The polarity_scores() function is used to find the sentiment intensity score for the tweet. This function gives various scores of which we have used “compound” metric to find the sentiment. Its value ranges from -1 to 1. So, if the value is greater than 0, it is classified as positive and if the value is less than 0, it is classified as negative. Lastly, if the value is equal to 0, it is classified as neutral. Lastly, we find the number of positive and negative tweets for Obama and Romney. This is a simple method that returns the results as 46.8% favoring Obama and 53.2% favoring Romney.

Method 2:

Similar to the previous method, we used the extracted tweets that contain Obama and Romney. Then, all the hyperlinks, RT symbols, and punctuation were removed from the tweet. For the second method, we used TfidfVectorizer from the sklearn.feature_extraction.text module to extract the features with ngram_range = (1,2) for word analyzer. Additionally, we removed the stop words from the tweets with TfidfVectorizer. This results in a matrix that is the vector representation of all the tweets. Then, using K-Means from the sklearn module, 2 clusters of positive and negative sentiments were formed. Lastly, after calculating the number of positive and negative tweets for Obama and Romney, we got 66% favoring Obama and 34% favoring Romney. We used this unsupervised clustering method because the dataset does not have labels.

Method 3:



As mentioned previously, in the preprocessing step, we extract the tweets with Obama and Romney and remove the hyperlinks, RT symbols, and punctuation from the tweet since they are not useful in finding the sentiment.

For the third method, the adjectives, adverbs, and verbs in a sentence are given high importance because they prove to be helpful in generating the sentiment of a sentence. So, we extracted the adjectives, adverbs and verbs for each tweet using POS tagger from the nltk library. After extraction, SentimentIntensityAnalyzer from the nltk.sentiment.vader module was used to score each word from the extracted words. The polarity_scores() function with “compound” metric is

used to score those extracted words. Now, the following process is used to rescore the extracted words:

- ➔ If the score of a verb or adverb is less than 0, it is negative and the following adjective score is subtracted from 5.
- ➔ Else if the score of a verb or adverb is greater than 0, it is positive and the score of the adjective that follows is multiplied to the score of that verb or adverb.

Next, the final score of the tweet is calculated by taking the sum of the new scores of the extracted words and divided by the number of adjectives in the tweets. Then, the last step is finding the number of positive and negative tweets for Obama and Romney. The result of this method was 53% favoring Obama and 47% favoring Romney.

Method 4:

As mentioned in the previous methods, after preprocessing, the tweets that included a mention of Obama and Romney were extracted. Next, using the a library in nltk, all the stop words were removed, such as ‘we’ or ‘our.’ However, a single modification was made to the list of stop words, which included ‘not.’ Since negation is a component in determining the sentiment in this method, that word had to be removed from the list. For this method, we extracted a text file that contained a list of positive and negative words provided from the University of Illinois. This list of words included commonly misspelled words that appeared frequently in social media. Using these lists, each tweet was evaluated against the words. If there were more positive words in the tweet, it was classified as positive. On the opposite hand, if there were more negative words, the tweet was categorized as negative. For cases where there were an equal number of positive and negative tweets, the tweet was considered to be neutral. Based on the number of positive tweets, the number of negative tweets, and the subject of the tweet, the final sentiment was classified into four categories: positive for Obama, negative for Obama, positive for Romney, and negative for Romney. The result of this method was 41.1% favoring Obama and 58.8% favoring Romney.

Method 5:

As a part of preprocessing, the id and text of the tweet were extracted from the dataset. Further extraction was done and we were left with the tweets that mention the presidential candidates in the text. The data is then processed further by removing punctuations, tokenizing the tweets text, stemming the words and then combining words from each tweet in stemmed form. The original tweets are then replaced with stemmed tweets in a dataframe. TextBlob library is used to get sentiments from each tweet, to classify them into three categories: positive, negative, and neutral. TextBlob gives sentiment of each of the tweets as a polarity score between [-1,1]. Negative polarity scores are assigned -1, positive scores are assigned +1, and neutral scores are assigned as zero. Using the tweets and assigned sentiments, we calculate whether the tweet is towards Obama or Romney. Finally, the number of positive tweets about Obama and negative tweets about Romney are added together to calculate the total number of tweets favoring Obama. Likewise, the number of positive tweets about Romney and negative tweets about Obama are added together to get the total number of tweets favoring Romney. The result of this method was 57.1% favoring Obama and 42.9% favoring Romney.

2.2 Technologies & Tools

Conda
Conda is an environment manager, as well as, package manager that can be used on various platforms. We used Conda for installing packages and libraries that were required for our project. Additionally, Conda manages the project dependencies, which was useful because we created different environments in which we ran our code.

Python
Python is a minimalistic and intuitive programming language that is perfect for machine learning projects. We used Python as it provided various useful libraries for analysis and processing data.

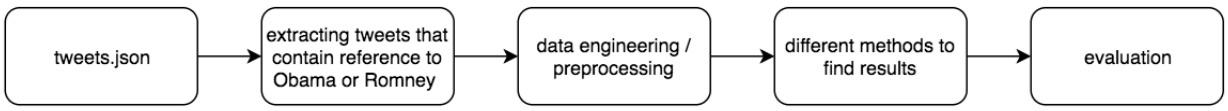
Jupyter Notebook
Jupyter Notebook is an open-source web application that can be used for generating files that contain code or text. We used Jupyter Notebook in our project to write the code for data analysis, preprocessing, and generating the results. The advantage of Jupyter Notebook is each block of code can be run sequentially but, separately. This feature aids in debugging and quick code changes.

Kaggle
We also used Kaggle environment to run our code and use its processing power since the dataset is very large.

Libraries

- SciPy is an open-source Python library that is used for scientific and technical computing.
- NumPy is a Python library that can support high-dimensional matrices and arrays. It also provides mathematical functions that can be operated on these arrays.
- Pandas is an open-source library for Python that includes easy to use data structures, as well as, data analysis tools.
- NLTK, or Natural Language Toolkit, is a platform that has many text processing libraries and is used for natural language processing. We used the tokenization, tagging, parsing, and semantic analysis libraries.
- Scikit-learn is an open-source machine learning library that provides both unsupervised and supervised algorithms.
- Matplotlib is a plotting library that is used by various other libraries, such as SciPy and NumPy.
- Plotly is used for data analytics and data visualization.
- Seaborn is based on Matplotlib and is another visualization library.
- TextBlob is a Python library for processing textual data.

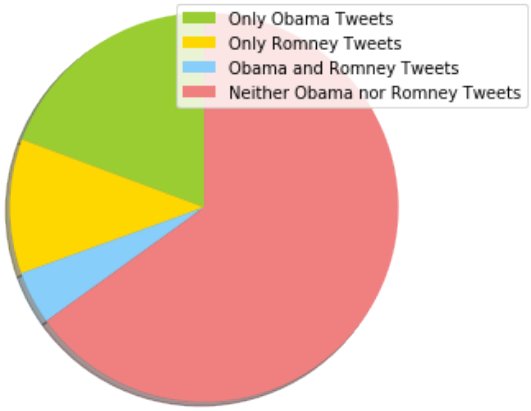
2.3 System Workflow

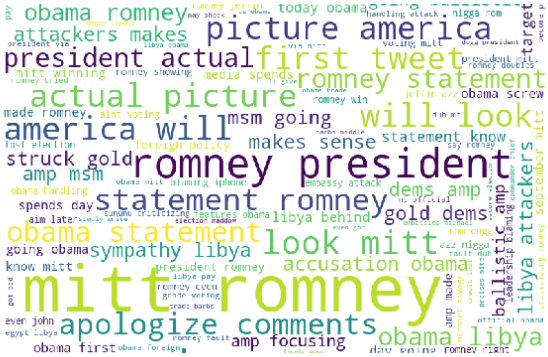


2.4 Visualization

We used data visualization for analysing and better understanding of the dataset.

The pie chart to the right represents the tweet distribution for Obama and Romney. The tweets that we used to determine the results are the 'Only Obama tweets' and 'Only Romney tweets.'





The two word clouds above are for the tweets containing Obama (left) and tweets containing Romney (right). Based on the word clouds, we can determine which words are commonly used in tweets.

3. Experiments

3.1 Dataset

The "Twitter 2012 USA presidential election" dataset that we have used is from Kaggle.

- Link : https://www.kaggle.com/jgoodman8/twitter-2012-presidential-election
- The zip file is of 470MB and the extracted full data is of size 3GB.
- The data is in JSON format.
- It consists of 1 million entries and it is multi-level JSON object that has information regarding the tweets and user.
- Each entry consists of 20 different properties/features. Few of those properties contain other JSON objects as values.

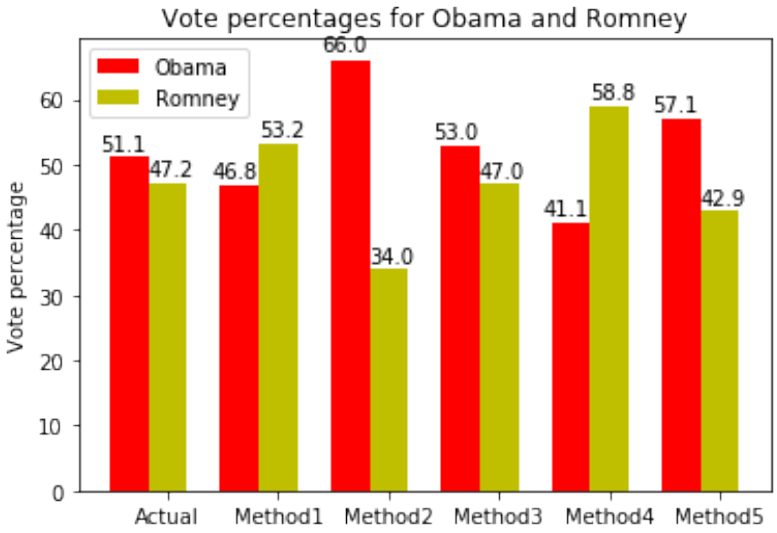
3.2 Data Preprocessing

- The features that are important to us for the sentiment analysis are the text of the tweet, hashtags used, and usernames mentioned.

- These features were extracted and the other features, such as created_at, source, user_id, and others, were removed because they were not useful.
- Empty or null tweets were removed as they were not useful in determining sentiment.
- Used a regular expression to remove the hashtag symbol (#) and username mention symbol (@).
- Moreover, during the data analysis we found the number tweets containing Obama or Romney are around 350K. We extracted the tweets with Obama and Romney referenced and removed the other tweets.

3.3 Results

- The graph below shows the comparison of the results of the methods mentioned previously.
- It can be seen that the graph represents the vote percentage of Obama and Romney that we got for the five different methods, as well as, the actual results.



3.4 Analysis of Results

- The actual percentage of the votes for Obama is 51.1% and Romney is 47.2%.
- **Best Method:** It can be observed that method 3 with the result as 53% for Obama and 47% for Romney is the most accurate.
- **Worst Method:** It can be observed that method 2 with the result as 66% for Obama and 34% for Romney is the least accurate.
- Although methods 1 and 4 inaccurately predict the overall outcome of the election, the individual percentages are less than 10% different from the actual percentages.
- Method 5 accurately predict the overall outcome of the election. Looking at the individual percentages, the difference between the predicted and actual is less than 7%.

4. Discussion & Conclusion

4.1 Decisions Made

- For preprocessing, removing all unnecessary features and only including text, hashtags, and usernames
- Discussed various methods that we were implementing to classify tweets as positive, negative, or neutral

4.2 Difficulties Faced

- The first problem we faced was figuring out what we can do with this Twitter dataset, which was unlabeled. We decided to determine the sentiment of the tweets and classify them into 4 different categories: positive for Obama, negative for Obama, positive for Romney, and negative for Romney. Using this information, we would be able to calculate the percentage of Twitter users that favored Obama and Romney.
- We had trouble determining the sentiment for the tweets that included both names. During preprocessing, we were able to find and filter tweets that included a mention of the candidates. If only one candidate was mentioned, it was straightforward in finding the sentiment of the tweet and determining if it was something positive or negative about the candidate. However, in cases when both candidates are mentioned, we were not able to figure out how to split the tweet and determine which candidate was being favored. Thus, for the scope of this project, we decided to filter out the tweets that either did not mention a name or contained both names.
- Running the program and parsing through a million tweets took a really long time.

4.3 Things That Worked

- We followed a similar approach in preprocessing with extracting the tweets that contained a reference to Obama or Romney.
- We all had different approaches to achieving the results.
- We all used Jupyter Notebook, which was helpful with dependencies and debugging.

4.4 Things That Didn’t Work

- Running the code on a million tweets took a really long time.

4.5 Conclusion

After evaluating our results, we found that the third method was the closest to the actual results. The actual percentages were 51.1% for Obama and 47.2% for Romney. The percentages from the third method were 53.0% for Obama and 47% for Romney. Of the remaining methods, only the second and fifth method accurately predicted who would win the presidential elections. The first and fourth method inaccurately predicted the overall result of the election. However, individually comparing the results from the election, the first, third, fourth, and fifth method predicted the percentage within 10% of the actual percentage of people who voted for Obama. Similarly, the first, third, and fifth method predicted the percentage within 10% of the actual percentage of people who voted for Romney.

The major lessons learned from this project are that preprocessing and feature engineering is an important step in sentiment analysis to get proper results. Also, it is difficult to check the accuracy when the data is unlabeled.

5. Project Plan & Task Distribution

After our goal for the project was established, we discussed the different preprocessing steps that we can follow. As a result, we used similar approaches in preprocessing. We decided to implement our approaches separately before coming together and discussing them. Thus, the result of this project is multiple approaches to determining the sentiment of each tweet. We tried to keep each method as different as possible.

We were all assigned to work on the data preparation, data preprocessing, and algorithms separately. However, we made sure to check in with each other during each phase. The first three methods were implemented by Anisha. Anjani created the fourth method. Lastly, Mrunali worked on the fifth method. The report was split into 3 parts and we all completed our parts before going over it together.

Task	**Subtask**	**Assignee**
Project Topic		All
Data Analysis		All
Data Preprocessing		All
Methods	1, 2, 3	Anisha
	4	Anjani
	5	Mrunali
Report		All
Presentation		All

6. References

https://repository.ihu.edu.gr/xmlui/bitstream/handle/11544/15204/UpdatedDissertation.pdf?sequence=1

https://medium.freecodecamp.org/how-to-build-a-twitter-sentiments-analyzer-in-python-using-textblob-948e1e8aae14

https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html

https://textblob.readthedocs.io/en/dev/quickstart.html#sentiment-analysis

https://www.analyticsvidhya.com/blog/2018/07/hands-on-sentiment-analysis-dataset-python/