**Description of the datasets**

In our work we use a tweets form Donald Trump President of America on different subjects. We extracted tweets using twitter API and manually set sentiment to three distinct classes positive, negative and neutral.

**Preprocessing**

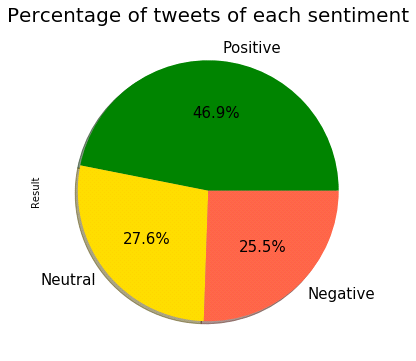
The dataset contain in csv file. We have extracted Tweets, created\_at, retweet\_count, favorite\_count, is\_retweet, Result columns from tweets. The total number of positive, negative and neutral tweets is shown in table.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **No. of tweets** | **Positive** | **Negative** | **Neutral** |
| **Dataset** | 5,703 | 2,674 | 1,457 | 1,572 |

Table: Statistics of the Twitter datasets used in the experiments

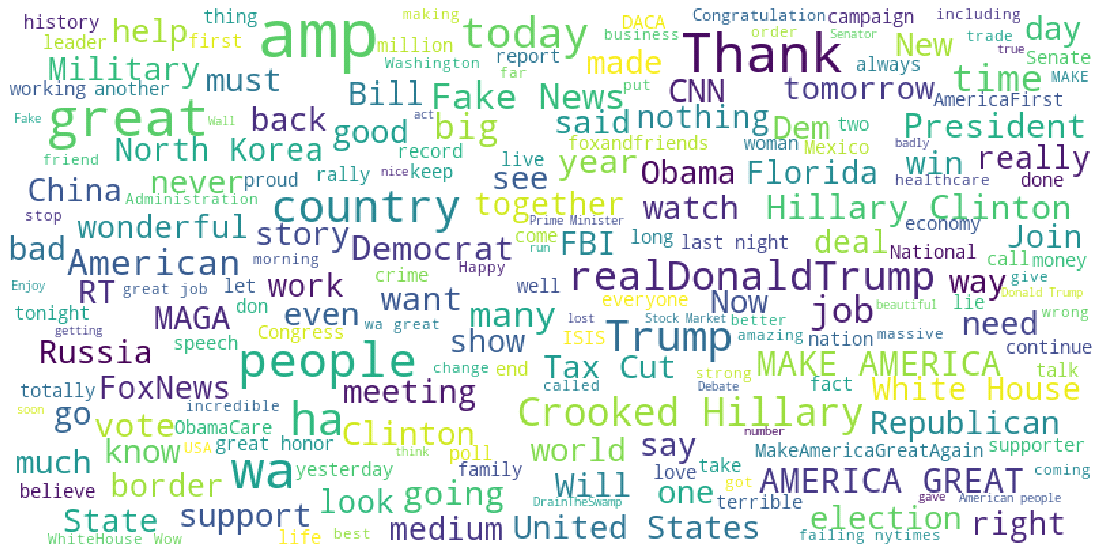
Total number of Percentage of tweets of each sentiment is shown in Figure

We can also remove commonly occurring words from the text data. The logic is that the commonly used words usually do not contribute to finding relevant differences and nuances in text data. Looking at the contents of tweets, removing common words from the analysis would not exclude many significant words, except for perhaps "I", appearing over 1100 times in c. 5700 tweets.

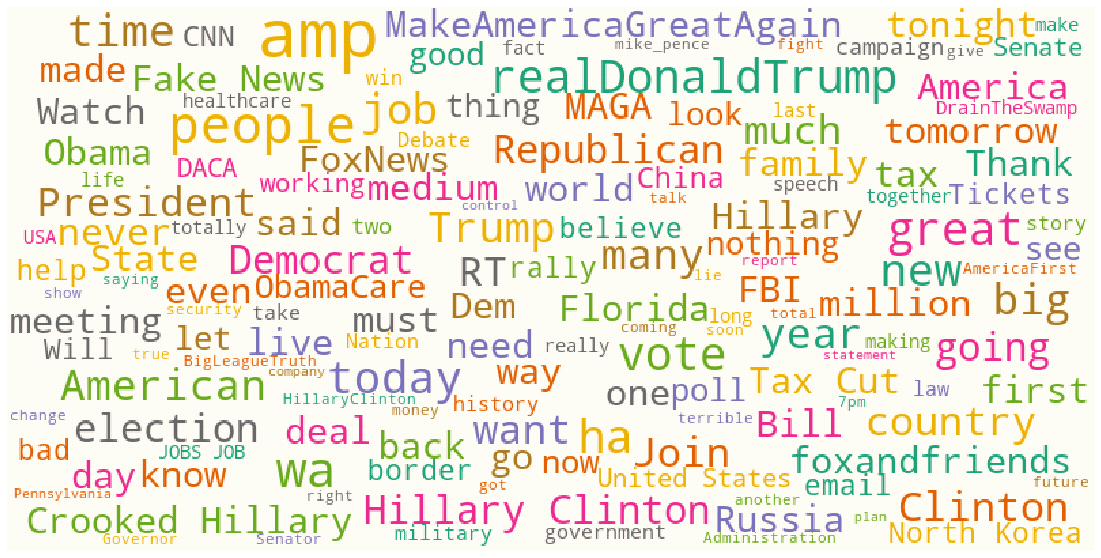


Total percentages of each sentiment.

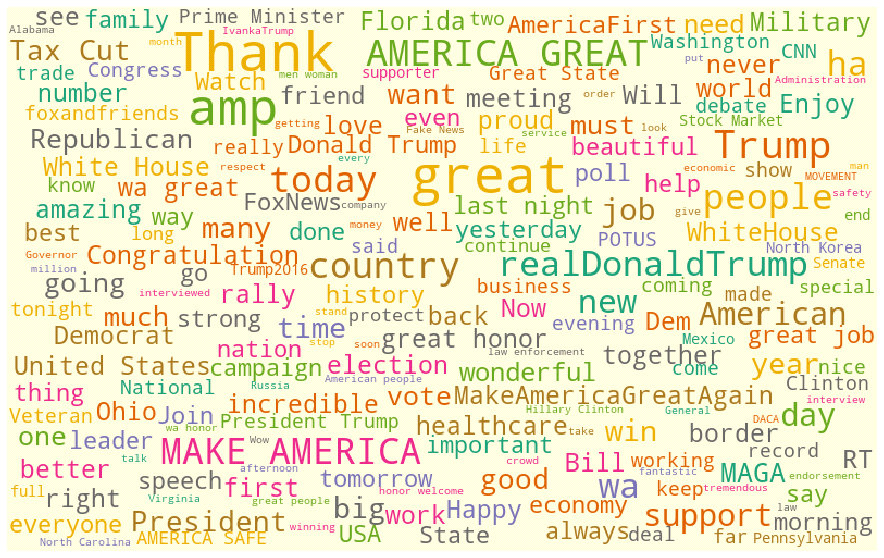
Similarly, just as we removed the most common words, maybe rarely occurring words should also be from the text. Because they’re so rare, the association between them and other words is dominated by noise. However, I will just assume that their impact on the final results would only be marginal.



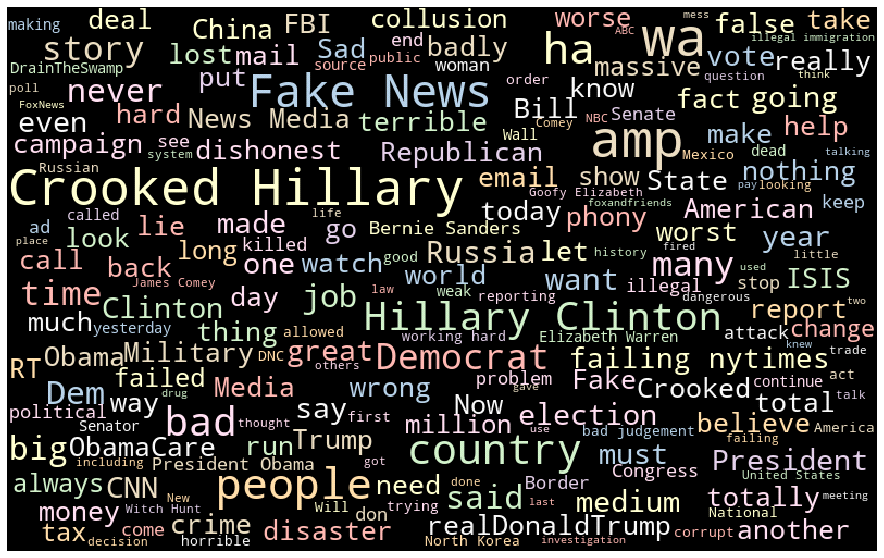
World cloud for all words in tweets



World cloud for Neutral tweets



World cloud for Positive tweets



World cloud for Positive tweets

We separate dataset in two different part train and test set by 30%. In which 70% is used for train data and 30% is for test data.

We then used TF-IDF Vectorizer to find how important a word in document is in comparison to the dataset.