ASSIGNMENT 3

BUSINESS INSIGHT REPORT

DATA

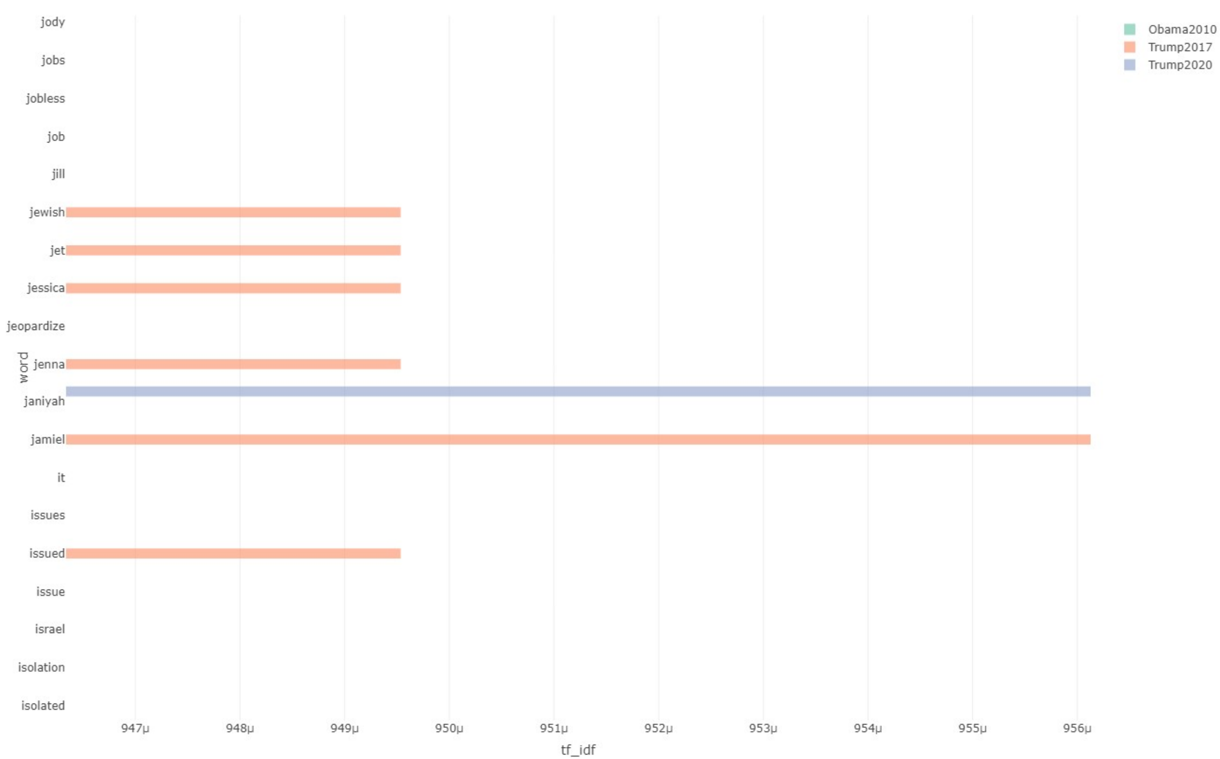
1. STATE OF THE UNION SPEECH BY PRESIDENT OBAMA, 2010
2. STATE OF THE UNION SPEECH BY PRESIDENT TRUMP, 2017
3. STATE OF THE UNION SPEECH BY PRESIDENT TRUMP, 2020

ANALYSIS

The State of the Union address is a speech delivered by the President of the United States to a joint session of the United States Congress. This speech usually happens at the start of each calendar year in office. The address generally includes a budget message, the economic report of the nation, and lets the President present and propose a legislative agenda and national priorities. The data for this analysis is the transcripts of this speech as delivered by Presidents' Obama and Trump in the year 2010, 2017 and 2020 respectively. The motivation of this analysis is to try and differentiate President Trump and President Obama by the basis of their speeches, try to find the different ideas their administration was inclined towards for that year. Also, to find out what were the similarities in their national agendas, if any. For the analysis, to obtain insights, frameworks such as tokenization, frequency, correlograms, word-clouds, sentiment analysis were used.

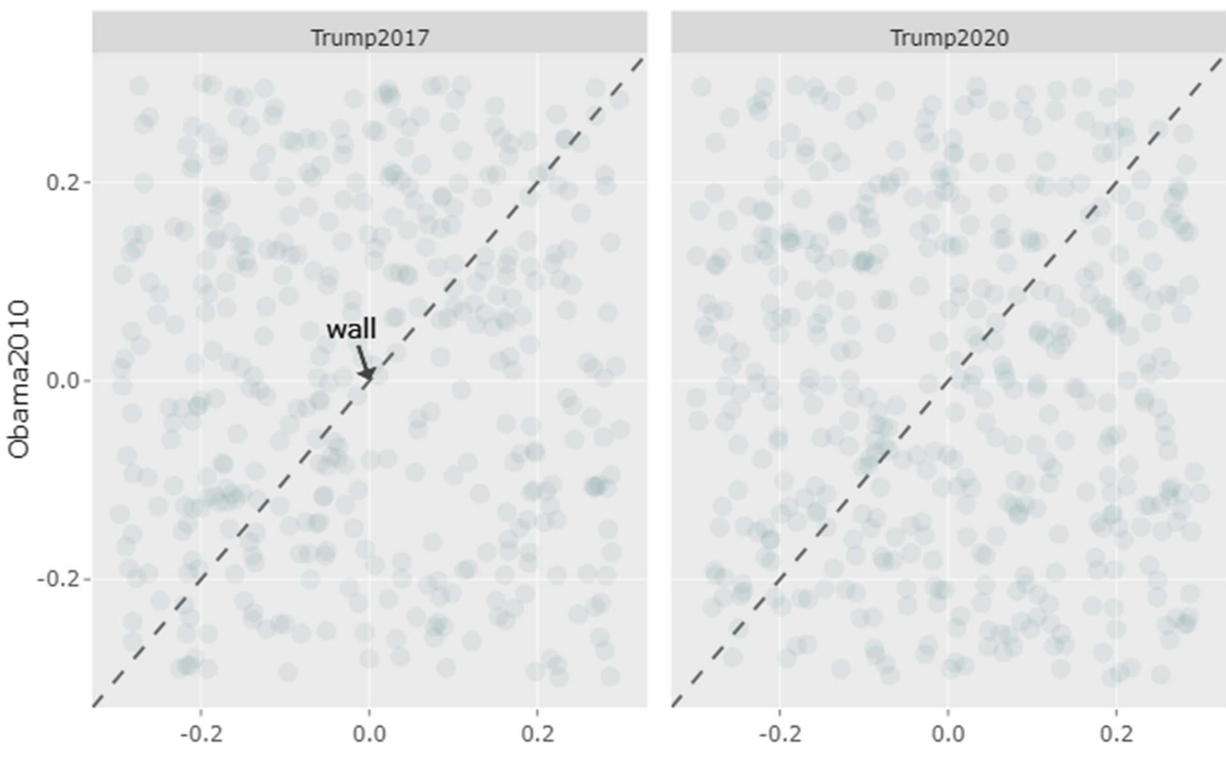
It is interesting to look at the token frequencies, not because they offer distinct insights but to notice how all three speeches used similar words. This shows that even both Presidents had different political agendas, their speeches were very presidential. “American”, “People”, are the most used words in all three speeches respectively. This does not offer any insights.

Looking at it from the opposite angle, as to which were the least used tokens, a tf_idf matrix is created and plotted. An interesting insight from this was that during both his speeches in 2017 and 2020, President Trump spoke and gave anecdotes about couple of American citizens. This can be said as, from the tf_idf plot, the names – “Jamiel”, “Jenna”, “Jessica” and “Janiyah” have the highest tf_idf value. On the other hand, if we look at the plot for President Obama, the names are not there. This is a distinction in the speeches where one has anecdotes and the other doesn’t.

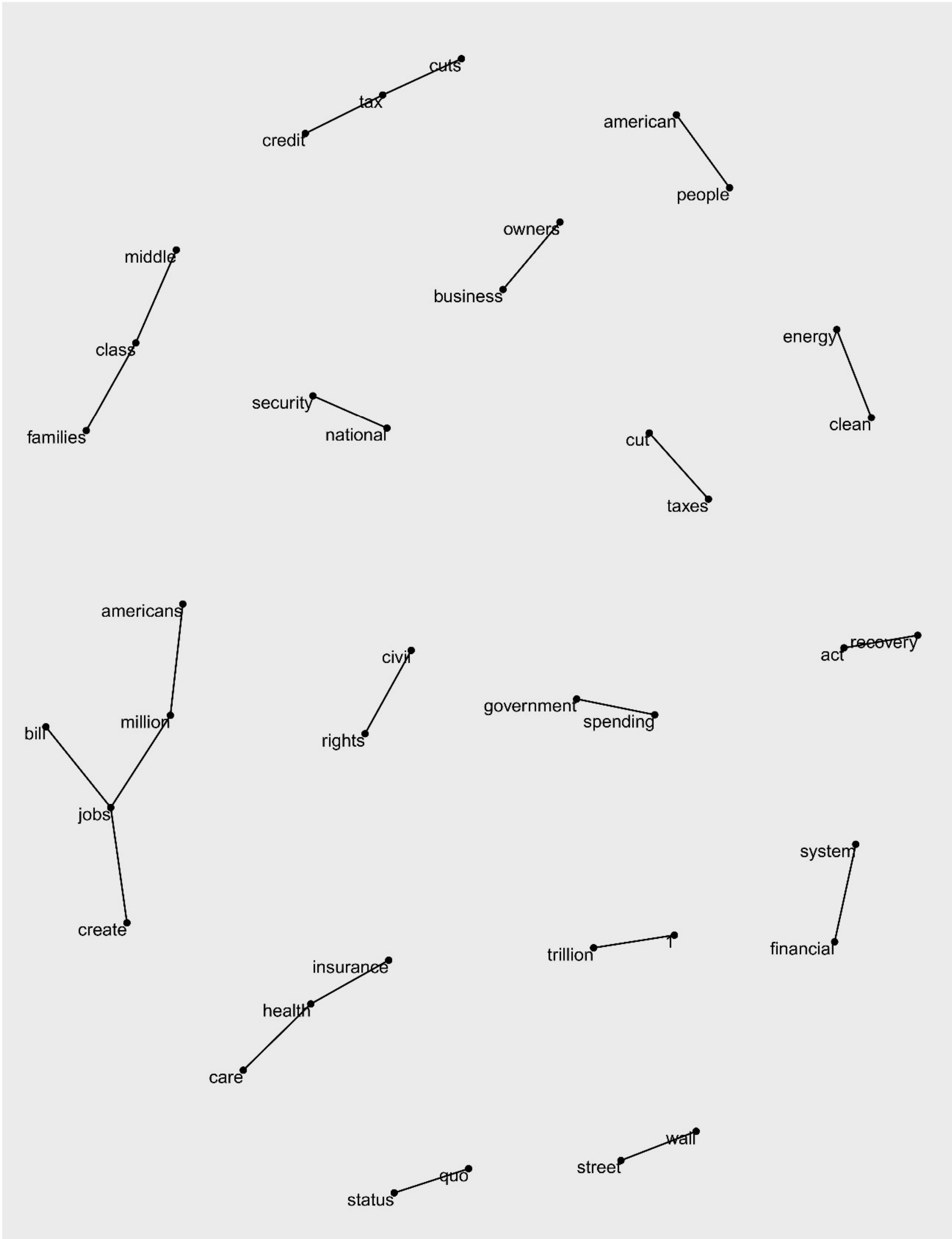


These being State of the Union speeches, should ideally not be very differentiable from the tokens as they are also used as a powerful political weapon to gather crowd support and increase their political ratings. It generally caters to the population at large and tend to be very similarly worded.

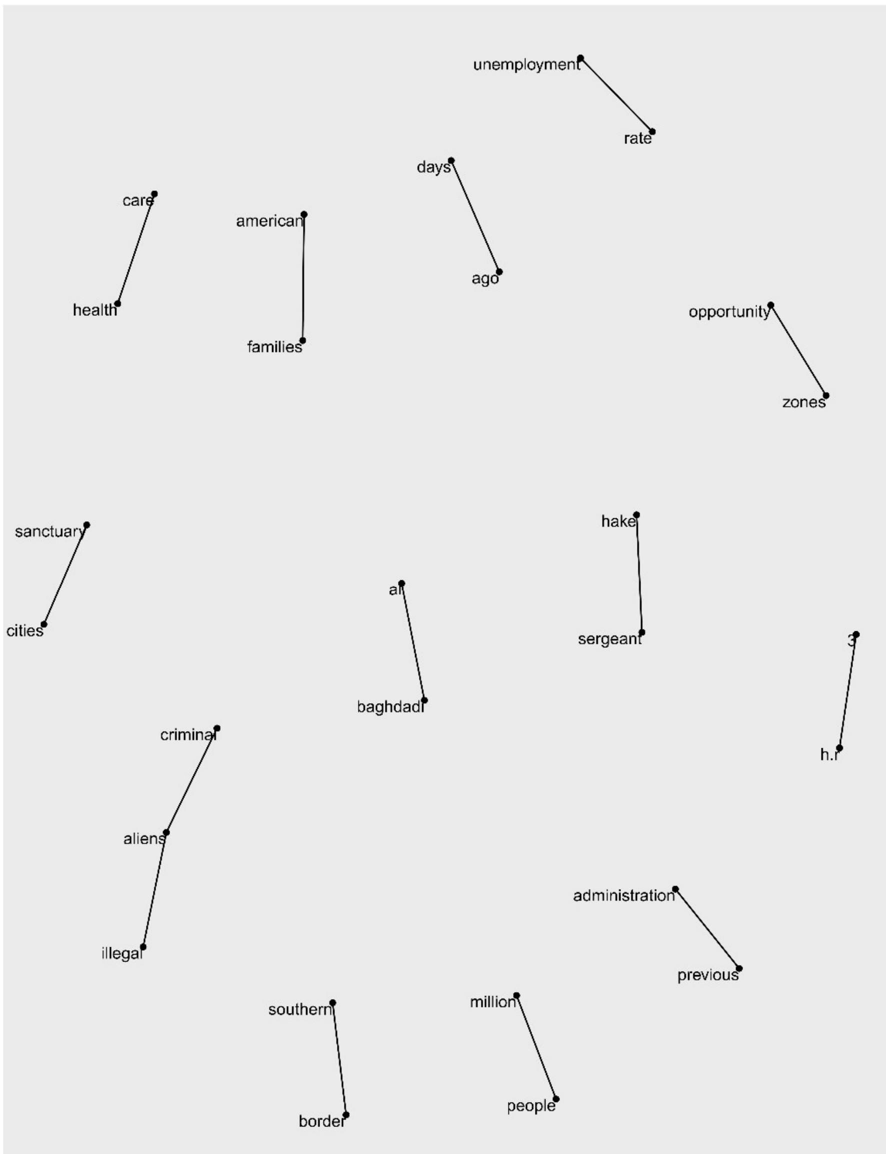
To look for further distinction, the data was tokenized into bigrams to see their usage. One of the most common words between President Trump's and President Obama's speeches was the word "wall". This can be seen in the below correlogram.



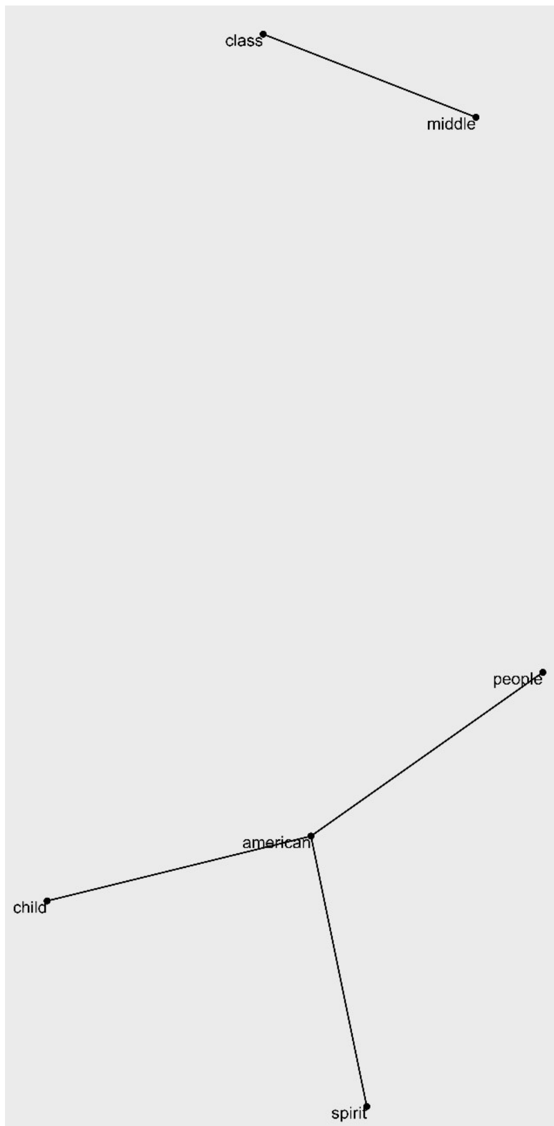
The interesting thing here is that President Trump talks with reference to the wall he wants to build at the border of Mexico and USA, where as President Obama talks with reference to "Wall Street". This can be seen from the bigrams. (below)



Apart from Wall Street, President Obama's address spoke about his policies such as health care, jobs bill, civil rights, clean energy and tax cuts. On the other hand, when this is looked at with what President Trump address, it is very different. This is insightful as when one is taking about either on of these policies, we can easily identify which administration is being referenced. President Trump addresses topics such as – Unemployment rate, al Baghdadi, illegal aliens, criminal aliens, southern border. (below)

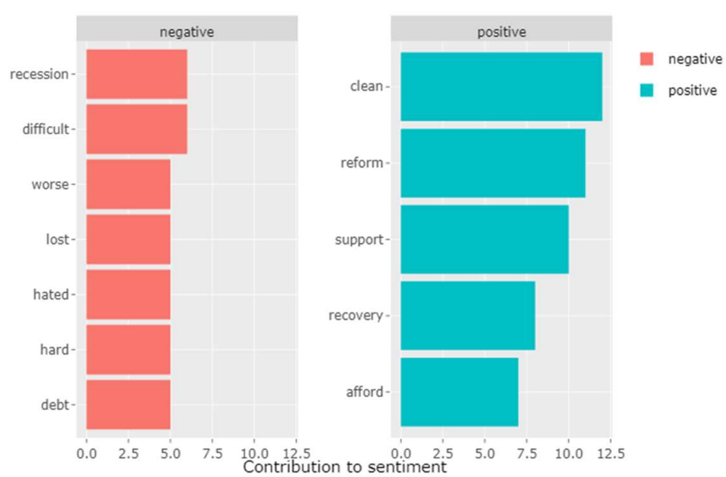


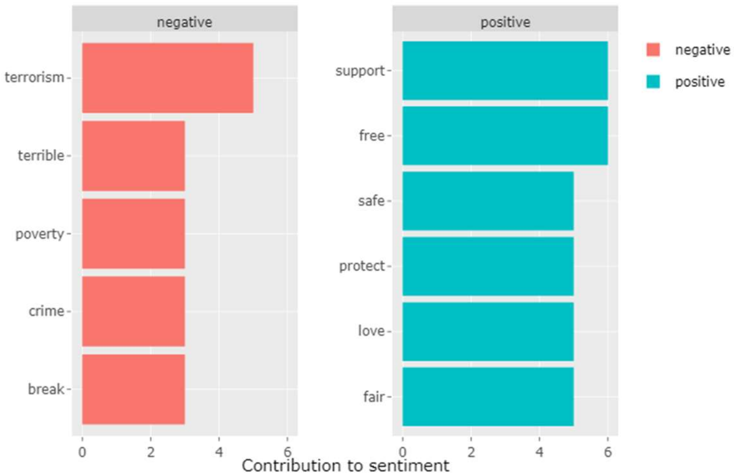
Another aspect to look at his how the administrations policies and view may have changed between their first term and last term. It is important as, during the first speech in 2017, President Trump would want to address the entire nation to rally more support as his victory was not well received. Whereas for his speech in 2020, he is trying to set a stronger foundation for his re-election so he may want to address his vote banks. In his 2017 speech, President Trump speaks on and about the American middle class, the American people and American spirit. Clearly showing that he is trying to rally bipartition support from the population and to increase his likeability. (below)

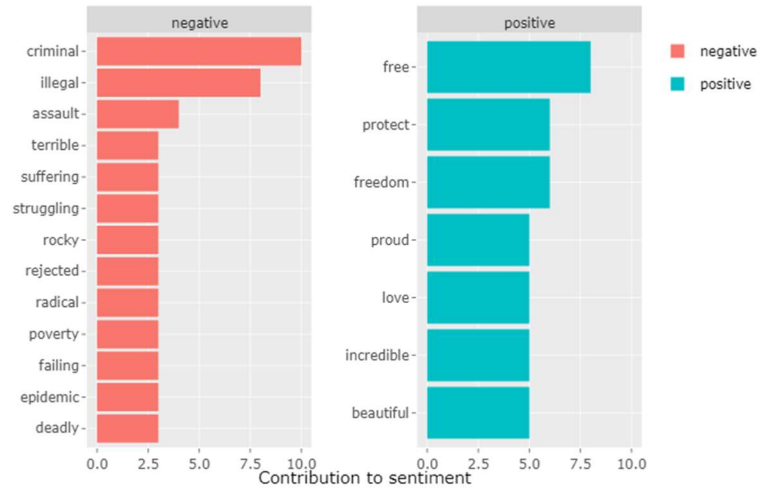


The three speeches can also be analyzed for their sentiments. This can be done so in three parts. Firstly, the speech as a whole – what it talks about, the language used or even the times and situations spoken about. Next, it can be analyzed by the token frequency of positive and negative tokens. And lastly, the analysis in terms of the moods and feeling it may portray. When the speeches are analyzed for the spear numbers, it shocking to find that the speech by President Obama is the one with the sentiment of -0.163 whereas President Trump's are 0.078 and 0.273 for 2017 and 2020, respectively. Although surprising, this isn't the full picture.

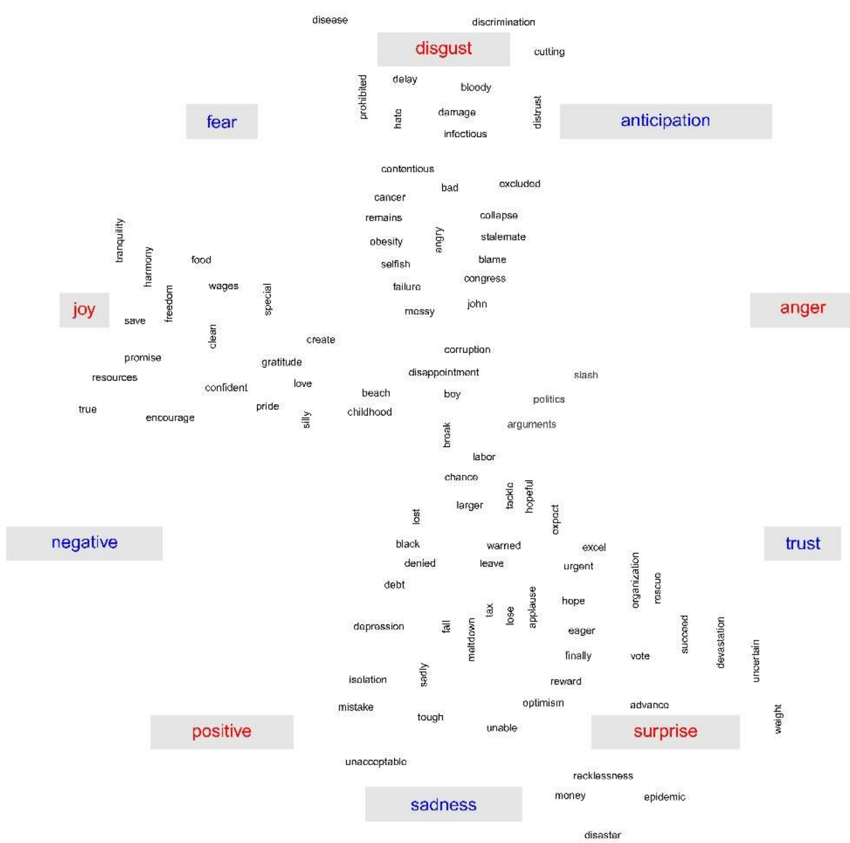
When we look at the positive and negative tokens that were used in the speeches, President Trump in his 2020 speech, used the negative tokens "Criminal" and "Illegal" most frequent but his speech also compensated highly with positive words such as "Free", "Protect", "Freedom" and "Proud". In his 2017 speech, there was a much more equal distribution of tokens across positive and negative. When we look at President Obama's speech from 2010, his positive tokens of "Clean", "Reform" and "Support" quickly get negated by the content of his speech where he spoke of the economy being in a bad state, and the tough and hard times that the American population had to face at the time. Below are the plots displaying the token frequencies by their sentiments. (in order, Obama 2010, Trump 2017, Trump 2020)(below)



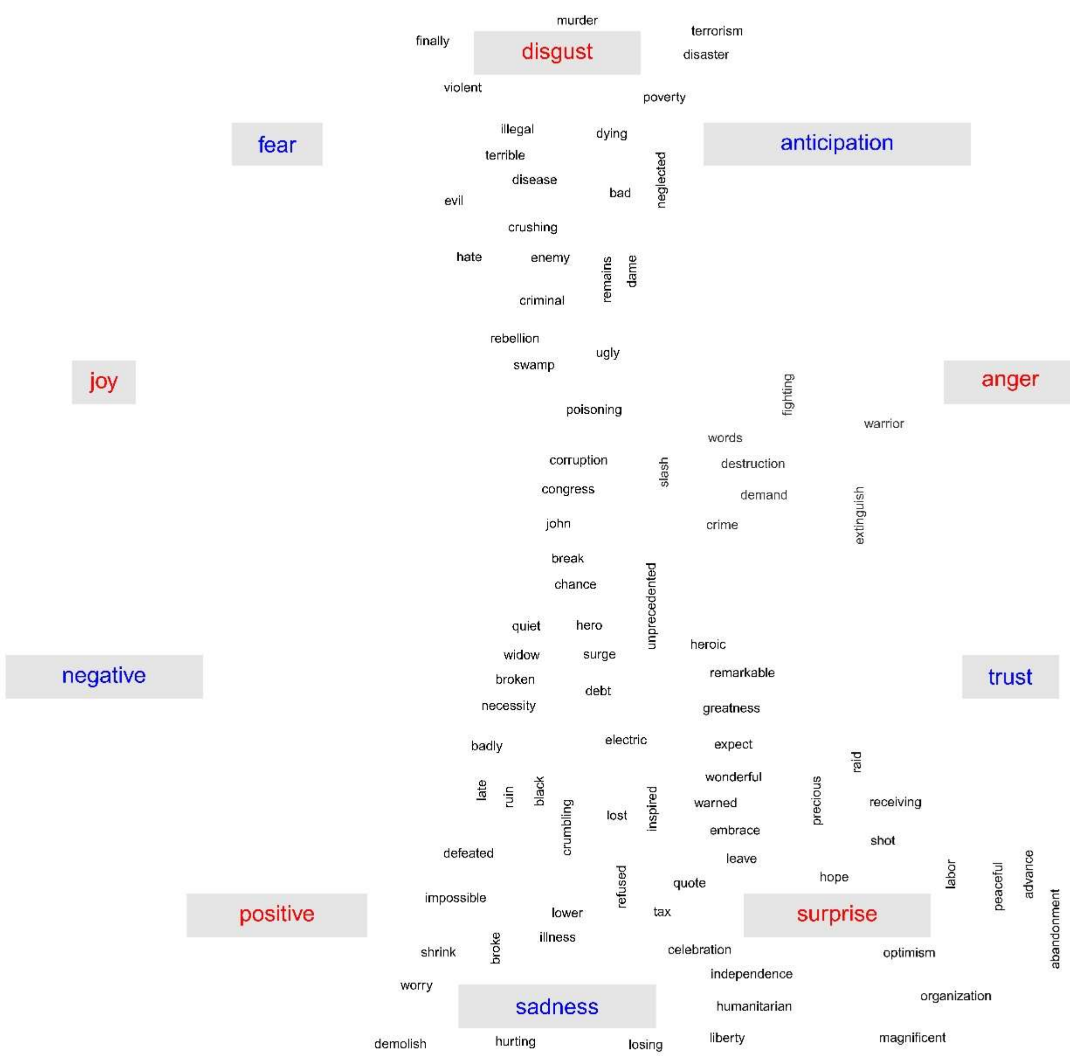




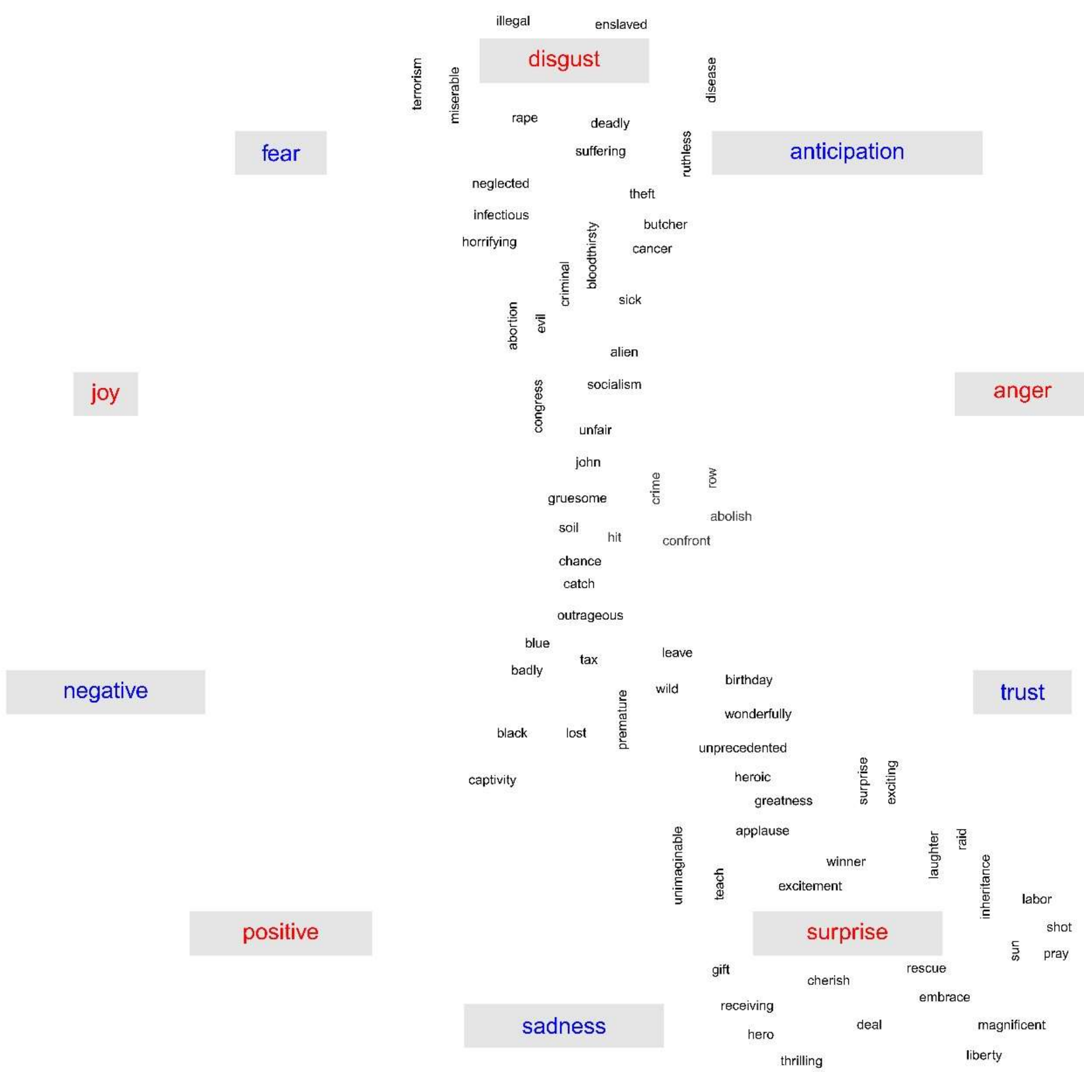
When doing sentiment analysis, we may also investigate the feelings it may portray. Since we already know from domain knowledge that the speeches talk about the times they were delivered in and the situation of the nation and its administration, a few interesting insights can be found from here. Below is the sentiment analysis for President Obama's speech from 2010.



From this analysis, by looking at the trends in the word-cloud, we can say that there was a feeling of joy and positivity. We also see trends for surprise and that may be since President Obama was the first colored President. The disgust and sadness may be due to the fact of what he spoke of in his speech. This can be said by looking at the words that are present in those trends. When we look at President Trump's from 2017, we see a similar trend for disgust and sadness due to what he speaks about. But we also notice that the trend for joy has now become a trend towards anger. Once again, the surprise trend may be due to the fact the it was his first State of the Union in an election which was one of the closest ever. Below is the word-cloud.



President Trump's 2020 speech which is focused on his re-election campaign when analyzed shows something like what we expect. The sentiments of anger have reduced as seen by a very small trend towards it. The sadness trend is missing, which would be since he wants to portray a happier time today than it was 4 years ago when he was elected for the first time. Below is the word-cloud.



In conclusion, the three speeches were very similar on the first glance but when analyzed further they are starkly different. One may expect that the two speeches by President Trump may be similar but from the analysis, that was not the case. It can clearly be identified which speech is it or which President we are talking about just by looking at these plots and not even have to see the speeches. The distinction is there which may be missed by most. When the Document Term Matrix is created for these speeches, we may see that the sparsity is only 54%. Ending with the example of President Obama speaking of "American Innovation" whereas President Trump speaks of "American Killings". (plot part of R Code).

PART 2 – R CODE

```
# Importing Packages

library(textdata)
library(tidytext)
library(tidyverse)
library(dplyr)
library(janeaustenr)
library(wordcloud)
library(textdata)
library(gutenbergr)
library(reshape2)
library(textreadr)
library(scales)
library(plotly)
library(igraph)
library(ggraph)
library(tm)
library(RColorBrewer)

# Reading Speech 1
file_1 <- read_document(file="Trump 2017.docx")
trump_2017 <- c(file_1)
trump_2017 <- data_frame(line=1, text=trump_2017)

# Tokenizing Speech 1
trump_2017_token <- trump_2017 %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words) %>%
  count(word, sort = TRUE)
```

trump_2017_token

```
> # Tokenizing Speech 1
> trump_2017_token <- trump_2017 %>%
+   unnest_tokens(word, text) %>%
+   anti_join(stop_words) %>%
+   count(word, sort = TRUE)
Joining, by = "word"
>
> trump_2017_token
# A tibble: 1,157 x 2
   word          n
   <chr>     <int>
 1 american     33
 2 america      27
 3 country      21
 4 world        17
 5 americans    15
 6 people       15
 7 united       14
 8 nation       13
 9 tonight      13
10 citizens     11
# ... with 1,147 more rows
>
```

```
# Reading Speech 2
file_2 <- read_document(file="Trump 2020.docx")
trump_2020 <- c(file_2)
trump_2020 <- data_frame(line=1, text=trump_2020)

# Tokenizing Speech 2
trump_2020_token <- trump_2020 %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words) %>%
  count(word, sort = TRUE)

trump_2020_token
```

```
> # Tokenizing Speech 2
> trump_2020_token <- trump_2020 %>%
+   unnest_tokens(word, text) %>%
+   anti_join(stop_words) %>%
+   count(word, sort = TRUE)
Joining, by = "word"
>
> trump_2020_token
# A tibble: 1,517 x 2
   word               n
   <chr>          <int>
 1 american          36
 2 people            22
 3 america           21
 4 country           21
 5 americans         20
 6 administration    19
 7 tonight           18
 8 america's         17
 9 care              16
10 world             15
# ... with 1,507 more rows
>
```

```
# Reading Speech 3
file_3 <- read_document(file="Obama 2010.docx")
obama_2010 <- c(file_3)
obama_2010 <- data_frame(line=1, text=obama_2010)

# Tokenizing Speech 3
obama_2010_token <- obama_2010 %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words) %>%
  count(word, sort = TRUE)
```

obama_2010_token

```
> # Tokenizing Speech 3
> obama_2010_token <- obama_2010 %>%
+   unnest_tokens(word, text) %>%
+   anti_join(stop_words) %>%
+   count(word, sort = TRUE)
Joining, by = "word"
>
> obama_2010_token
# A tibble: 1,360 x 2
   word           n
   <chr>      <int>
 1 people        32
 2 americans     28
 3 jobs          23
 4 time          19
 5 america       18
 6 american      18
 7 businesses    18
 8 families      17
 9 economy       15
10 energy        15
# ... with 1,350 more rows
>
```

```
# Combining into single data frame
speech <- bind_rows(mutate(trump_2020_token, author="Trump2020"),
                mutate(trump_2017_token, author= "Trump2017"),
                mutate(obama_2010_token, author="Obama2010"))%>%#closing bind_rows
 mutate(word=str_extract(word, "[a-z']+")) %>%
 count(author, word) %>%
 group_by(author)
```

speech

```
> speech
# A tibble: 3,886 x 3
# Groups:   author [3]
   author    word            n
   <chr>     <chr>       <int>
 1 Obama2010 abide           1
 2 Obama2010 ability         1
 3 Obama2010 abroad          1
 4 Obama2010 absolutely      1
 5 Obama2010 abuses          1
 6 Obama2010 accept          1
 7 Obama2010 access          1
 8 Obama2010 account         1
 9 Obama2010 accountable     1
10 Obama2010 achievement     1
# ... with 3,876 more rows
```

```
# Creating IDF
speech_idf <- speech %>%
  bind_tf_idf(word, author, n)

speech_idf
```

```
> speech_idf
# A tibble: 3,886 x 6
# Groups:   author [3]
   author    word             n       tf   idf   tf_idf
   <chr>     <chr>        <int>    <dbl> <dbl>    <dbl>
 1 Obama2010 abide            1 0.000735 1.10  0.000808
 2 Obama2010 ability          1 0.000735 0     0
 3 Obama2010 abroad           1 0.000735 0.405 0.000298
 4 Obama2010 absolutely       1 0.000735 0.405 0.000298
 5 Obama2010 abuses           1 0.000735 1.10  0.000808
 6 Obama2010 accept           1 0.000735 1.10  0.000808
 7 Obama2010 access           1 0.000735 0     0
 8 Obama2010 account          1 0.000735 1.10  0.000808
 9 Obama2010 accountable      1 0.000735 1.10  0.000808
10 Obama2010 achievement      1 0.000735 1.10  0.000808
# ... with 3,876 more rows
```

```
# Plotting IDF using Plotly
t <- plot_ly(data = speech_idf, x=~tf_idf, y=~word, color =~author, opacity = 0.6)
t

# Bing Sentiment Analysis using Plotly
# Trump 2020
trump2020_senti <- trump_2020 %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words) %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort=T) %>%
  ungroup()

trump2020_bing <- trump2020_senti %>%
  group_by(sentiment) %>%
  top_n(5) %>%
  ungroup() %>%
  mutate(word=reorder(word, n)) %>%
  ggplot(aes(word, n, fill=sentiment)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~sentiment, scales = "free_y")+
  labs(y="Contribution to sentiment", x=NULL)+
  coord_flip()

trump2020_bing <- ggplotly(trump2020_bing)
trump2020_bing
```

```
# Trump 2017
trump2017_senti <- trump_2017 %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words) %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort=T) %>%
  ungroup()

trump2017_bing <- trump2017_senti %>%
  group_by(sentiment) %>%
  top_n(5) %>%
  ungroup() %>%
  mutate(word=reorder(word, n)) %>%
  ggplot(aes(word, n, fill=sentiment)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~sentiment, scales = "free_y")+
  labs(y="Contribution to sentiment", x=NULL)+
  coord_flip()

trump2017_bing <- ggplotly(trump2017_bing)
trump2017_bing

# Obama 2010
obama_20210_senti <- obama_2010 %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words) %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort=T) %>%
  ungroup()

obama_2010_bing <- obama_20210_senti %>%
  group_by(sentiment) %>%
  top_n(5) %>%
  ungroup() %>%
  mutate(word=reorder(word, n)) %>%
  ggplot(aes(word, n, fill=sentiment)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~sentiment, scales = "free_y")+
  labs(y="Contribution to sentiment", x=NULL)+
  coord_flip()

obama_2010_bing <- ggplotly(obama_2010_bing)
obama_2010_bing
```

```
# NRC Sentiment Analysis - Word Cloud
# Trump 2020
trump2020_senti_nrc <- trump_2020 %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words) %>%
  inner_join(get_sentiments("nrc")) %>%
  count(word, sentiment, sort=T) %>%
  ungroup()

trump2020_senti_nrc %>%
  inner_join(get_sentiments("nrc")) %>%
  count(word, sentiment, sort=TRUE) %>%
  acast(word ~sentiment, value.var="n", fill=0) %>%
  comparison.cloud(colors = c("grey20", "gray80"),
                   title.colors=c("red","blue"),
                   max.words=100, fixed.asp=TRUE,
                   scale=c(0.6,0.6), title.size=1, rot.per=0.25)

# Trump 2017
trump_2017_senti_nrc <- trump_2017 %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words) %>%
  inner_join(get_sentiments("nrc")) %>%
  count(word, sentiment, sort=T) %>%
  ungroup()

trump_2017_senti_nrc %>%
  inner_join(get_sentiments("nrc")) %>%
  count(word, sentiment, sort=TRUE) %>%
  acast(word ~sentiment, value.var="n", fill=0) %>%
  comparison.cloud(colors = c("grey20", "gray80"),
                   title.colors=c("red","blue"),
                   max.words=100, fixed.asp=TRUE,
                   scale=c(0.6,0.6), title.size=1, rot.per=0.25)

# Obama 2010
obama_2010_senti_nrc <- obama_2010 %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words) %>%
  inner_join(get_sentiments("nrc")) %>%
  count(word, sentiment, sort=T) %>%
  ungroup()

obama_2010_senti_nrc %>%
  inner_join(get_sentiments("nrc")) %>%
  count(word, sentiment, sort=TRUE) %>%
  acast(word ~sentiment, value.var="n", fill=0) %>%
  comparison.cloud(colors = c("grey20", "gray80"),
```

```
        title.colors=c("red","blue"),
        max.words=100, fixed.asp=TRUE,
        scale=c(0.6,0.6), title.size=1, rot.per=0.25)


# Creating Bigrams and plotting the networks
# Trump 2020
trump2020_bigrams <- trump_2020 %>%
  unnest_tokens(bigram, text, token = "ngrams", n=2) %>%
  separate(bigram, c("word1", "word2"), sep = " ") %>%
  filter(!word1 %in% stop_words$word) %>%
  filter(!word2 %in% stop_words$word) %>%
  count(word1, word2, sort = TRUE)

trump2020_bigrams
```

```
> trump2020_bigrams
# A tibble: 1,042 x 3
   word1        word2         n
   <chr>        <chr>     <int>
 1 health       care          9
 2 unemployment rate          7
 3 days         ago           5
 4 illegal      aliens        4
 5 al           baghdadi      3
 6 american     families      3
 7 criminal     aliens        3
 8 h.r          3             3
 9 million      people        3
10 opportunity  zones         3
# ... with 1,032 more rows
>
```

```
trump2020_bigram_graph <- trump2020_bigrams %>%
  filter(n>2) %>%
  graph_from_data_frame()

ggraph(trump2020_bigram_graph, layout = "fr") +
  geom_edge_link()+
  geom_node_point()+
  geom_node_text(aes(label=name),  vjust =1, hjust=1)

# Trump 2017
trump2017_bigrams <- trump_2017 %>%
  unnest_tokens(bigram, text, token = "ngrams", n=2) %>%
  separate(bigram, c("word1", "word2"), sep = " ") %>%
  filter(!word1 %in% stop_words$word) %>%
  filter(!word2 %in% stop_words$word) %>%
  count(word1, word2, sort = TRUE)
```

trump2017_bigrams

```
> trump2017_bigrams
# A tibble: 630 x 3
   word1    word2         n
   <chr>    <chr>     <int>
 1 health   insurance     4
 2 american child         3
 3 american people        3
 4 american spirit        3
 5 middle   class         3
 6 43       million       2
 7 6        trillion      2
 8 american companies     2
 9 american greatness     2
10 audience tonight       2
# ... with 620 more rows
>
```

```
trump2017_bigram_graph <- trump2017_bigrams %>%
  filter(n>2) %>%
  graph_from_data_frame()

ggraph(trump2017_bigram_graph, layout = "fr") +
  geom_edge_link()+
  geom_node_point()+
  geom_node_text(aes(label=name),  vjust =1, hjust=1)

# Obama 2010
obama2010_bigrams <- obama_2010 %>%
  unnest_tokens(bigram, text, token = "ngrams", n=2) %>%
  separate(bigram, c("word1", "word2"), sep = " ") %>%
  filter(!word1 %in% stop_words$word) %>%
  filter(!word2 %in% stop_words$word) %>%
  count(word1, word2, sort = TRUE)
```

obama2010_bigrams

```
> obama2010_bigrams
# A tibble: 686 x 3
   word1    word2        n
   <chr>    <chr>    <int>
 1 american people      11
 2 clean    energy      10
 3 health   care         7
 4 cut      taxes        6
 5 recovery act          6
 6 tax      cuts         6
 7 middle   class        5
 8 1        trillion     4
 9 tax      credit       4
10 wall     street       4
# ... with 676 more rows
>
```

```
obama2010_bigram_graph <- obama2010_bigrams %>%
  filter(n>2) %>%
  graph_from_data_frame()

ggraph(obama2010_bigram_graph, layout = "fr") +
  geom_edge_link()+
  geom_node_point()+
  geom_node_text(aes(label=name),  vjust =1, hjust=1, scale=c(0.6,0.6))


# Creating single data frame with all tokens along with frequency proportions
frequency <- bind_rows(mutate(trump_2020_token, author="Trump2020"),
                       mutate(trump_2017_token, author= "Trump2017"),
                       mutate(obama_2010_token, author="Obama2010")
)%>%#closing bind_rows
  mutate(word=str_extract(word, "[a-z']+")) %>%
  count(author, word) %>%
  group_by(author) %>%
  mutate(proportion = n/sum(n))%>%
  #select(-n) %>%
  spread(author, proportion) %>%
  gather(author, proportion, `Trump2020`, `Trump2017`)

frequency
```

```
> frequency
# A tibble: 5,614 x 5
   word              n Obama2010 author    proportion
   <chr>         <int>     <dbl> <chr>          <dbl>
 1 abandon           1 NA        Trump2020   0.000659
 2 abandonment       1 NA        Trump2020  NA
 3 abide             1  0.000735 Trump2020  NA
 4 abiding           1 NA        Trump2020   0.000659
 5 ability           1  0.000735 Trump2020   0.000659
 6 abolish           1 NA        Trump2020   0.000659
 7 abortion          1 NA        Trump2020   0.000659
 8 abraham           1 NA        Trump2020   0.000659
 9 abroad            1  0.000735 Trump2020  NA
10 absolutely        1  0.000735 Trump2020   0.000659
# ... with 5,604 more rows
```

```
# Plotting a correlogram using Plotly
p <- ggplot(frequency, aes(x=proportion, y=Obama2010,
                           color = abs(Obama2010- proportion)))+
  geom_abline(color="grey40", lty=2)+
  geom_jitter(aes(text=paste("word: ", word)), alpha=.1, size=2.5, width=0.3, height=0.3)+
  #geom_text(aes(label=word), colour="gray20", alpha=1) +
  #scale_x_log10(labels = percent_format())+
  #scale_y_log10(labels= percent_format())+
```

```
  scale_color_gradient(limits = c(0,0.001), low = "darkslategray4", high = "gray75")+
  facet_wrap(~author, ncol=2)+
  theme(legend.position = "none")+
  labs(y= "Obama2010", x=NULL)

p <- ggplotly(p)
p


# Creating a Document Term Matrix (DTM)
speech_dtm <- speech %>%
  group_by(author) %>%
  cast_dtm(author, word, n)

speech_dtm
```

```
> speech_dtm
<<DocumentTermMatrix (documents: 3, terms: 2787)>>
Non-/sparse entries: 3886/4475
Sparsity           : 54%
Maximal term length: NA
Weighting          : term frequency (tf)
>
```

```
# Creating a vector for negeation tokens
negation_tokens <- c('wall', 'student', 'job', 'american')

# Negeated Token for Trump 2020
trump2020_negated <- trump_2020 %>%
  unnest_tokens(bigram, text, token = "ngrams", n=2) %>%
  separate(bigram, c("word1", "word2"), sep = " ") %>%
  filter(!word1 %in% stop_words$word) %>%
  filter(!word2 %in% stop_words$word) %>%
  filter(word1 %in% negation_tokens) %>%
  inner_join(get_sentiments('afinn'), by=c(word2="word")) %>%
  count(word1, word2, value, sort=TRUE) %>%
  ungroup()

trump2020_negated

# Negeated Token for Trump 2017
trump2017_negated <- trump_2017 %>%
  unnest_tokens(bigram, text, token = "ngrams", n=2) %>%
  separate(bigram, c("word1", "word2"), sep = " ") %>%
  filter(!word1 %in% stop_words$word) %>%
  filter(!word2 %in% stop_words$word) %>%
  filter(word1 %in% negation_tokens) %>%
  inner_join(get_sentiments('afinn'), by=c(word2="word")) %>%
  count(word1, word2, value, sort=TRUE) %>%
```

```
  ungroup()

trump2017_negated

# Negeated Token for Obama 2010
obama2010_negated <- obama_2010 %>%
  unnest_tokens(bigram, text, token = "ngrams", n=2) %>%
  separate(bigram, c("word1", "word2"), sep = " ") %>%
  filter(!word1 %in% stop_words$word) %>%
  filter(!word2 %in% stop_words$word) %>%
  filter(word1 %in% negation_tokens) %>%
  inner_join(get_sentiments('afinn'), by=c(word2="word")) %>%
  count(word1, word2, value, sort=TRUE) %>%
  ungroup()

obama2010_negated

# Function to plot negeted tokens
negated_plot <- function(x){
  obama2010_negated %>%
    filter(word1 == x) %>%
    mutate(contribution = n* value) %>%
    arrange(desc(abs(contribution))) %>%
    head(20) %>%
    mutate(word2 = reorder(word2, contribution)) %>%
    ggplot(aes(word2, n*value, fill = n*value >0))+
    geom_col(show.legend = FALSE)+
    xlab(paste("Words preceded by", x))+
    ylab("Sentiment score* number of occurences")+
    coord_flip()
}

# Negeation plot using the plot fuction created above
negated_plot(x="job")
negated_plot(x="american")

# Sentiment Analysis using AFINN
# Trump 2020
trump2020_afinn <- trump_2020_token %>%
  inner_join(get_sentiments("afinn"))%>%
  summarise(mean(value))
```

trump2020_afinn

```
> trump2020_afinn
# A tibble: 1 x 1
  `mean(value)`
          <dbl>
1         0.273
>
```

```
# Trump 2017
trump2017_afinn <- trump_2017_token %>%
  inner_join(get_sentiments("afinn"))%>%
  summarise(mean(value))
```

trump2017_afinn

```
> trump2017_afinn
# A tibble: 1 x 1
  `mean(value)`
          <dbl>
1        0.0798
>
```

```
# Obama 2010
obama_2010_afinn <- obama_2010_token %>%
  inner_join(get_sentiments("afinn"))%>%
  summarise(mean(value))
```

obama_2010_afinn

```
> obama_2010_afinn
# A tibble: 1 x 1
  `mean(value)`
          <dbl>
1        -0.163
>
```