The Tone and Focus of the State of the Union Address

Alex McKeever

10/31/2023

```
#Loading necesary packages
library(readr)
library(tidyverse)
library(stringr)
library(knitr)
library(tidytext)
library(lubridate)
library(wordcloud)
library(reshape2)
library(dplyr)
```

```
#Loading the data
speech <- read_rds("stateofunion1790_2022_tibble.rds")
```

```
#Making all lower case as case does not really matter here
speech$text <- tolower(speech$text)
```

```
#Removing crowd reactions
speech$text <- speech$text %>%
  str_remove_all("\\[\\w+\\]")
```

```
#Removing punctuation
speech$text <- speech$text %>%
  str_remove_all('[\\.\\,\\?\\;\\:\\?\\(\\)\\"\\$]') %>%
  str_replace_all(pattern = "--", replacement = " ")
```

```
#Removing extra information that is record keeping and not part of the speech
speech$text <- speech$text %>%
  str_remove_all(
    "the address as reported from the floor appears in the congressional.*")
```

```
#Removing double/triple spaces
speech$text <- str_squish(speech$text)
```

```
#Converting dates to a workable format
speech$date <- mdy(speech$date)
```

```
#Seperating speeches into three time frames
speech_time_1 <- speech %>%
  filter(between(date, as.Date("1790-01-01"), as.Date("1870-01-01")))
speech_time_2 <- speech %>%
  filter(between(date, as.Date("1870-01-02"), as.Date("1950-01-01")))
speech_time_3 <- speech %>%
  filter(between(date, as.Date("1950-01-02"), as.Date("2024-01-01")))
```

```
#Making word count for each time period and removing stop words (such as 'the')
#Using unnest_tokens to turn speeches into individual words to work with
speech %>%
  select(text) %>%
  unnest_tokens(output = word, input = text) %>%
  anti_join(stop_words) %>%
  count(word, sort = TRUE) %>%
  kable()
speech_time_1 %>%
  select(text) %>%
  unnest_tokens(output = word, input = text) %>%
  anti_join(stop_words) %>%
  count(word, sort = TRUE) %>%
  kable()
speech_time_2 %>%
  select(text) %>%
  unnest_tokens(output = word, input = text) %>%
  anti_join(stop_words) %>%
  count(word, sort = TRUE) %>%
  kable()
speech_time_3 %>%
  select(text) %>%
  unnest_tokens(output = word, input = text) %>%
  anti_join(stop_words) %>%
  count(word, sort = TRUE) %>%
  kable()
```

```
#Making word clouds for each seperate time period (and overall)
#Only using words with positive or negative sentiments with inner_join
#Once again, not using stop words
set.seed(1234)
speech %>%
  select(text) %>%
  unnest_tokens(output = word, input = text) %>%
  anti_join(stop_words) %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = TRUE) %>%
  acast(word ~ sentiment, value.var = "n", fill = 0) %>%
  comparison.cloud(colors = c("blue", "purple"),
 scale = c(2,0.5),
 max.words = 65,
 title.size = 2)
```

negative

positive

```
speech_time_1 %>%
  select(text) %>%
  unnest_tokens(output = word, input = text) %>%
  anti_join(stop_words) %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = TRUE) %>%
  acast(word ~ sentiment, value.var = "n", fill = 0) %>%
  comparison.cloud(colors = c("blue", "purple"),
 scale = c(2, 0.5),
 max.words = 65,
 title.size = 2)
```


```
speech_time_2 %>%
  select(text) %>%
  unnest_tokens(output = word, input= text) %>%
```

```
  anti_join(stop_words) %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = TRUE) %>%
  acast(word ~ sentiment, value.var = "n", fill = 0) %>%
  comparison.cloud(colors = c("blue", "purple"),
 scale = c(2, 0.5),
 max.words = 65,
 title.size = 2)
```

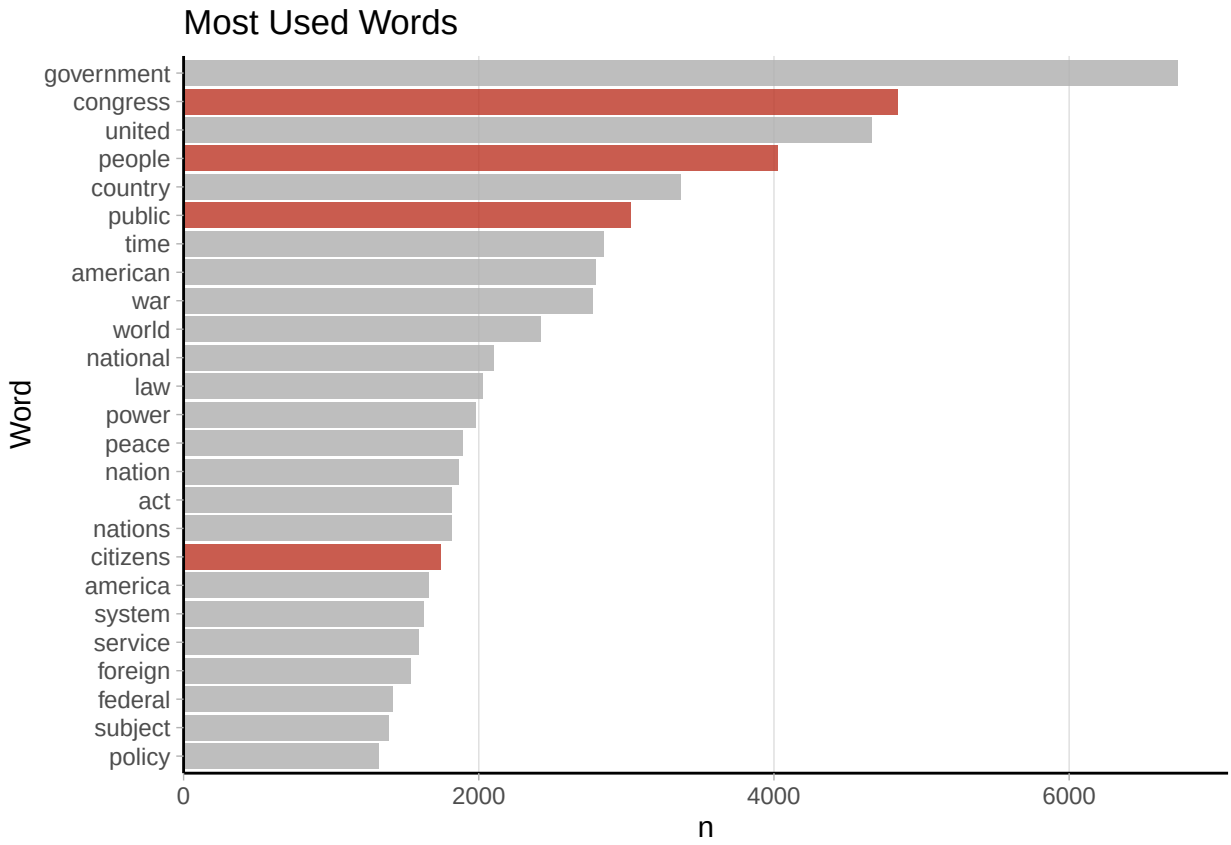

```
speech_time_3 %>%
  select(text) %>%
  unnest_tokens(output = word, input= text) %>%
  anti_join(stop_words) %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = TRUE) %>%
  acast(word ~ sentiment, value.var = "n", fill = 0) %>%
  comparison.cloud(colors = c("blue", "purple"),
 scale = c(2, 0.5),
 max.words = 50,
 title.size = 2)
```


```
#Counting total number of instances of each word in all speeches
#Using this data to create a bar chart of the most used words
#Audience indicating words are highlighted
```

```r
total_word_count <- speech %>%
  select(text) %>%
  unnest_tokens(output = word, input = text) %>%
  anti_join(stop_words) %>%
  count(word, sort = TRUE)

total_word_count$highlight <- total_word_count$word %in% c("congress", "people",
                                                           "public", "citizens")

total_word_count %>%
  slice(1:25) %>%
  ggplot() +
  geom_col(mapping = aes(x = fct_reorder(word, n), y = n,
                         fill = highlight)) +
  coord_flip() +
  theme_light() +
      scale_fill_manual(values = c("#B0B0B0D0", "#BD3828D0")) +
    theme(panel.grid.minor = element_blank(), panel.grid.major.y = element_blank(),
        panel.border = element_blank(),
        axis.line = element_line(color = 'black'),
        legend.position = "none") +
      scale_y_continuous(expand = expansion(mult = c(0, 0.05))) +
  labs(
    title = "Most Used Words",
    x = "Word"
  )
```



```r
#Calculating count of debt, congress, and people for each speech and joining it
#to the data frame
debt_count <- speech %>%
  group_by(date) %>%
  unnest_tokens(output = word, input = text) %>%
  anti_join(stop_words) %>%
  filter(word == "debt") %>%
  count(word, sort = TRUE)
speech <- speech %>%
  left_join(debt_count) %>%
  rename(debt = n) %>%
  select(-word)
speech[is.na(speech)] <- 0

congress_count <- speech %>%
  group_by(date) %>%
  unnest_tokens(output = word, input = text) %>%
  anti_join(stop_words) %>%
  filter(word == "congress") %>%
  count(word, sort = TRUE)
speech <- speech %>%
  left_join(congress_count) %>%
  rename(congress = n) %>%
  select(-word)
speech[is.na(speech)] <- 0

people_count <- speech %>%
  group_by(date) %>%
  unnest_tokens(output = word, input = text) %>%
  anti_join(stop_words) %>%
  filter(word == "people") %>%
  count(word, sort = TRUE)
speech <- speech %>%
  left_join(people_count) %>%
  rename(people = n) %>%
  select(-word)
speech[is.na(speech)] <- 0
```

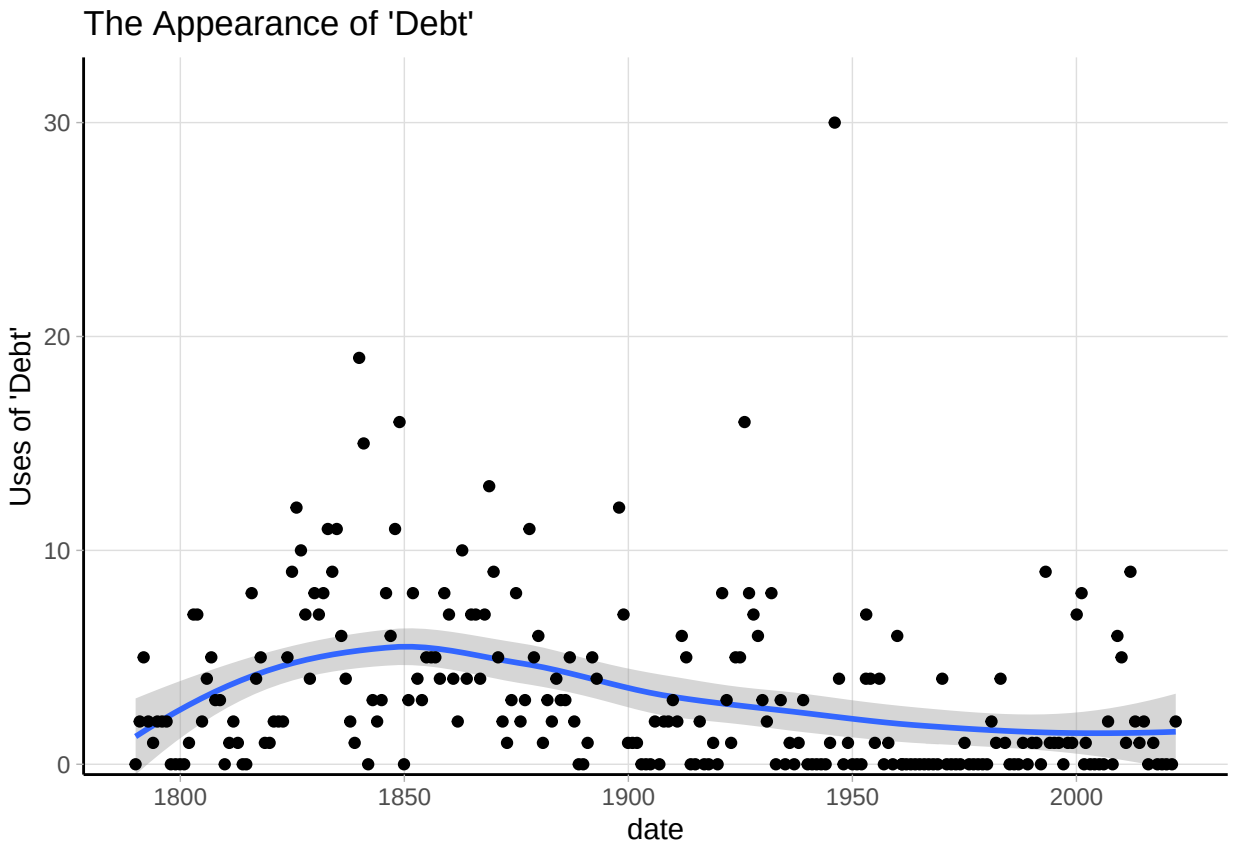
```r
#Counting up the total instances of positive or negative words for each speech and
#joining these counts to the original data frame
sentiment_count <- speech %>%
  group_by(date) %>%
  unnest_tokens(output = word, input = text) %>%
  anti_join(stop_words) %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = TRUE)
positive_count <- sentiment_count %>%
  filter(sentiment == "positive") %>%
  select(-word) %>%
  group_by(date) %>%
  summarise(positive = sum(n))
negative_count <- sentiment_count %>%
  filter(sentiment == "negative") %>%
```

```
  select(-word) %>%
  group_by(date) %>%
  summarise(negative = sum(n))
speech <- speech %>%
  left_join(positive_count)
speech <- speech %>%
  left_join(negative_count)
```

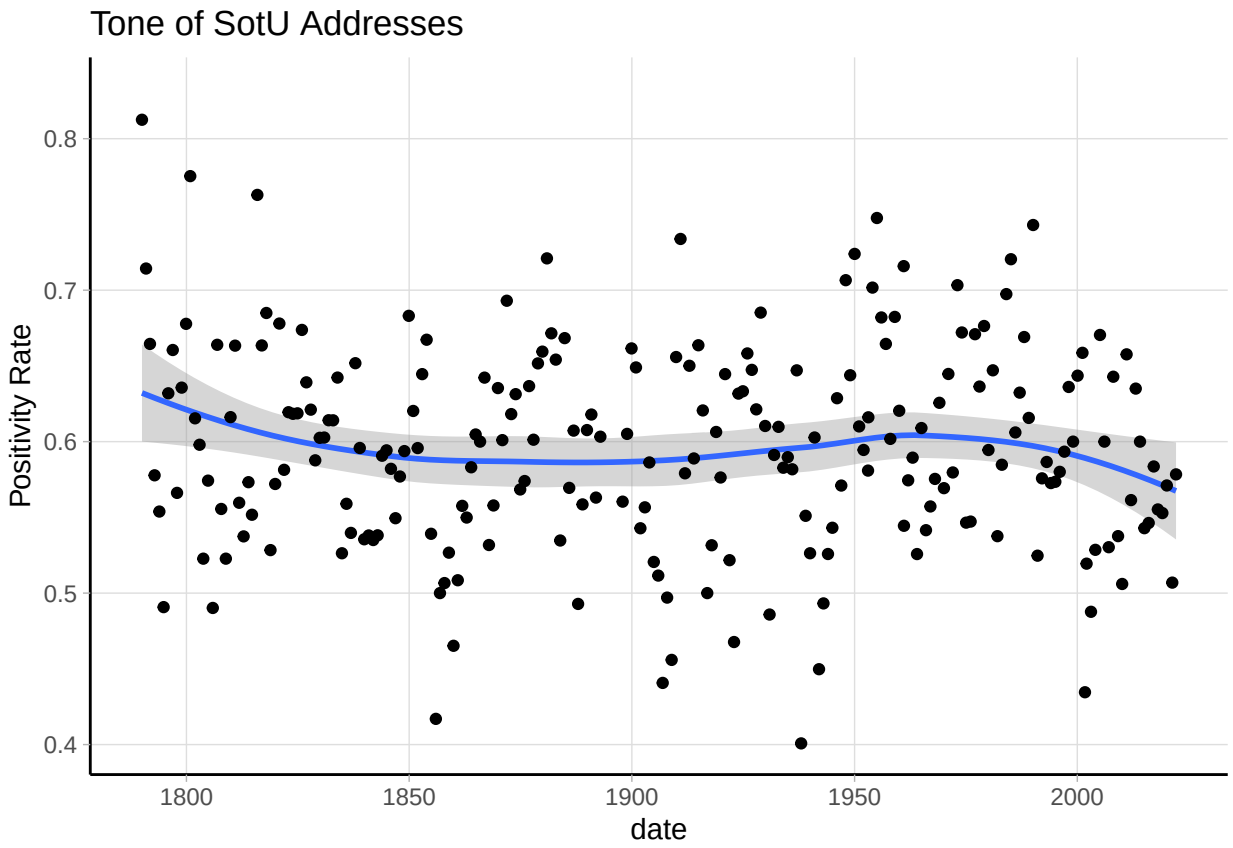
```
#Finding positivity rate as speeches tend to varry by length thus it is crucial
#to compare the percentage of positive words as opposed to raw count
speech <- speech %>%
  mutate(positive_rate = positive/(positive + negative))
```

```
#Making the data tidy and allowing the comparison of government and people in later ggplot
#Storing as seperate data frame as positive rate
#can still be useful in the original data frame
speech_pivot <- speech %>%
  pivot_longer(
    cols = "debt":"negative",
 names_to = "word",
 values_to = "count")
```

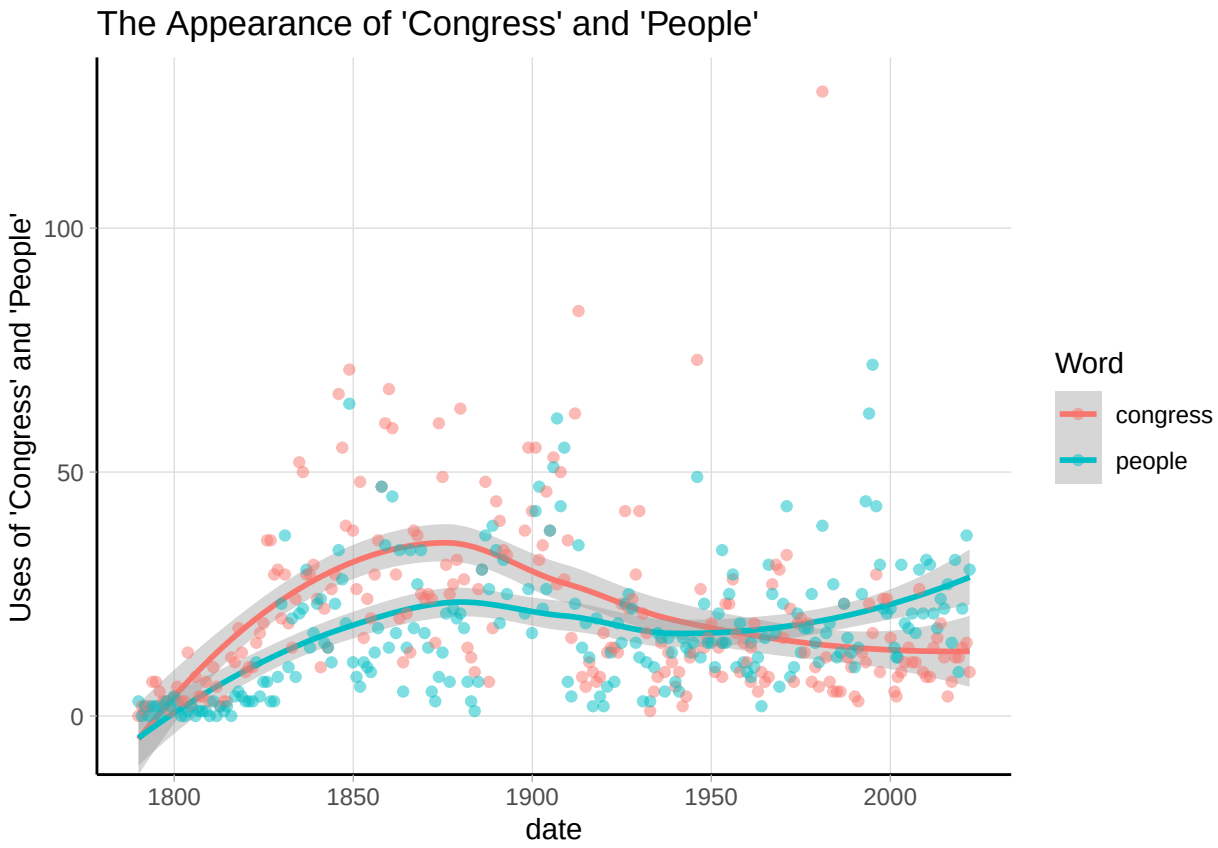
```
#Graph of debt appearances
speech_pivot %>%
  filter(word == "debt") %>%
  ggplot() +
  geom_smooth(aes(x = date, y = count)) +
  geom_point(aes(x = date, y = count)) +
    theme_light() +
    theme(panel.grid.minor = element_blank(),
        panel.border = element_blank(), axis.line = element_line(color = 'black')) +
    scale_y_continuous(expand = expansion(mult = c(0, 0.1))) +
  labs(
    y = "Uses of 'Debt'",
    title = "The Appearance of 'Debt'"
  )
```




```
#Graph of positive_rate for each speech
speech %>%
  ggplot() +
  geom_smooth(aes(x = date, y = positive_rate)) +
  geom_point(aes(x = date, y = positive_rate)) +
    theme_light() +
    theme(panel.grid.minor = element_blank(),
        panel.border = element_blank(), axis.line = element_line(color = 'black')) +
    scale_y_continuous(expand = expansion(mult = c(0.05, 0.1))) +
  labs(
    y = "Positivity Rate",
    title = "Tone of SotU Addresses"
  )
```




```
#Graph of congress and people appearances over time
speech_pivot %>%
  filter(word == "congress" | word == "people") %>%
  ggplot() +
  geom_smooth(aes(x = date, y = count, color = word)) +
  geom_point(aes(x = date, y = count, color = word), alpha = 0.5) +
    theme_light() +
    theme(panel.grid.minor = element_blank(),
        panel.border = element_blank(), axis.line = element_line(color = 'black')) +
    scale_y_continuous(expand = expansion(mult = c(0, 0.05))) +
  labs(
    y = "Uses of 'Congress' and 'People'",
    title = "The Appearance of 'Congress' and 'People'",
    color = "Word"
  )
```


The Appearance of 'Congress' and 'People'
Uses of 'Congress' and 'People'
100
50
0
1800
1850
1900
1950
2000
date
Word
congress
people