Narrative

Yi Lu

12/10/2020

Brief background

This project pertains to a brief analysis of US politicians' address on "social ism". "Socialism" as both an ideology and a practice that criticizes and seeks to overcome capitalist market economy has aroused considerable attack and negation from the mainstream politicians. Neither Democrats nor Republicans would publicly support socialism, but their concepts on and patterns of using the term "socialism"/"socialist" differed. Based on public documents retrieved from UCSB's American Presidency Project, this project attempts to show the diff- erent patterns of use of "socialism" as a politically sensitive term. I use plotting and wordcloud analysis to explore the data.

Collecting data

Getting data can be very painful especially when many of the raw materials are irregular in their formats and you cannot write a one-time-for-all code to retr- ieve them. In this project I use data scraped from the American Presidency Project by searching the keyword "socialism"

```
# Getting necessary urls from the searching pages is what we want, but first we
# need to prepare R packages and the searching pages.

library(tidyverse)
```

```
## -- Attaching packages --------------------------------------------------- tidyverse 1.3.0 --
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
## -- Conflicts ------------------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(rvest)
```

```
## Loading required package: xml2
##
## Attaching package: 'rvest'
## The following object is masked from 'package:purrr':
##
##     pluck
## The following object is masked from 'package:readr':
##
##     guess_encoding
```

```r
library(stringr)
library(purrr)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```r
library(tm)
```

```
## Loading required package: NLP
```

```
##
## Attaching package: 'NLP'
```

```
## The following object is masked from 'package:ggplot2':
##
##     annotate
```

```r
library(wordcloud)
```

```
## Loading required package: RColorBrewer
```

```r
# These are the results by searching keyword "socialism". I compile them into a vector.
no_socialism <- c(
"https://www.presidency.ucsb.edu/advanced-search?field-keywords=%22socialism%22&field-keywords2=&field-
"https://www.presidency.ucsb.edu/advanced-search?field-keywords=%22socialism%22&field-keywords2=&field-
"https://www.presidency.ucsb.edu/advanced-search?field-keywords=%22socialism%22&field-keywords2=&field-
"https://www.presidency.ucsb.edu/advanced-search?field-keywords=%22socialism%22&field-keywords2=&field-
"https://www.presidency.ucsb.edu/advanced-search?field-keywords=%22socialism%22&field-keywords2=&field-
"https://www.presidency.ucsb.edu/advanced-search?field-keywords=%22socialism%22&field-keywords2=&field-

# Write the csv of those unprocessed urls.
write.csv(no_socialism, "~/links.csv")
```

```r
# Create a scraping function to retrieve the urls we need (those with public remarks content).
stronk <- function(url){
  yes <- read_html(url)

  yes1 <- yes %>%
  html_nodes("a") %>%
  html_attr("href")

yes2 <- yes1[str_starts(yes1, "/documents")]
yes3 <- paste0("https://www.presidency.ucsb.edu", yes2)[-(1:4)]
  return(yes3)
}
```

```r
# Unfortunately those document pages do not contain the information about Presidents' political parties

party <- function(url){
  coup <- read_html(url)

  coup1 <- coup %>%
  html_nodes("a") %>%
```

```
  html_attr("href")

  coup2 <- coup1[str_starts(coup1, "/people")]

  coup3 <- paste0("https://www.presidency.ucsb.edu", coup2)[-1]
}
```

Processing data

This process is in fact not strikingly different from the previous one because now I am scraping the text information from those urls prepared in the previous step. Nevertheless we are going to process the raw urls to retrieve more useful information.

```
# Create the scraping functions to retrieve the information we need from the pages.

scrape_docs <- function(URL){
  doc <- read_html(URL)

  speaker <- html_nodes(doc, ".diet-title a") %>%
    html_text()

  date <- html_nodes(doc, ".date-display-single") %>%
    html_text() %>%
    mdy()

  text <- html_nodes(doc, "div.field-docs-content") %>%
    html_text()

  all_info <- list(speaker = speaker, date = date, text = text)

  return(all_info)
  # This returns the content of public remarks that have mentioned "socialism".
}

# Scraping from people's info pages.
obtain <- function(url){
  page <- read_html(url)

  party <- html_nodes(page, ".f-item") %>%
    html_text()

  get <- list(party = party[4])

  return(get)
  # This returns the political parties the POTUS affiliated with.
}
```

```
# Map the scraping function onto the vector we created.
maga <- map(no_socialism, stronk)

maga <- unlist(maga)

potus_stronk <- map_dfr(maga, scrape_docs)
```

```
potus_party <- map(no_socialism, party)

potus_party <- unlist(potus_party)

party_name <- map_dfr(potus_party, obtain)

new_potus <- data.frame(party_name, potus_stronk)

# Write the csv file of the processed data.
write.csv(new_potus, "~/potus.csv", row.names = F)
```

Analysis and Visualization

Now the raw data is processed, we are ready to analyze it with special tools of visualization.

```
# We'd like to see how many of public remarks related to "socialism" had been made from the period amid
num <- new_potus %>%
  count(date > "2008-1-30")
pie(num$n, labels = num$`date > "2008-1-30"`, clockwise = T, main = "Public remarks after 2008")

# For comparison, we would like to see how many public remarks were made before Cold War formally ended
num1 <- new_potus %>%
  count(date < "1991-12-27")
pie(num1$n, labels = num1$`date < "1991-12-27"`, clockwise = T, main = "Public remarks before 1991")
# Discussion of "socialism" among politicians is much more frequent after 2008, as the data shows that
```

```
# We now want to see the 10 politicians who talked most about "socialism"
num_speaker <- new_potus %>%
  count(speaker) %>%
  slice_max(n, n=10)
view(num_speaker)
# Guess who talks the most about "socialism"? Donald J. Trump! By October 2020 he had referred to "soci

# Using ggplot() to visualize the results
ggplot(data = num_speaker, aes(x = reorder(speaker, n), y = n)) + geom_bar(stat = "identity") + theme(a
angle = 45, size = 9, hjust = 1))+xlab("Speakers")+ylab("Number of docs")+
  ggtitle("10 speakers who talked the most about socialism")

## If we know nothing about politicians at all, we would think that Trump really "loves" socialism!
```

```
# We want to see with and without Trump, how Republicans' frequency of talking publicly about socialism
num2 <- new_potus %>%
  count(party == "Republican")
pie(num2$n, labels = num2$`party == "Republican"`, clockwise = T, main = "Public remarks made by GOP Pr
# With Trump included, Republican Presidents made public comments on socialism as frequently as their D

num3 <- new_potus %>%
  filter(!(speaker == "Donald J. Trump")) %>%
  count(party == "Republican")
pie(num3$n, labels = num3$`party == "Republican"`, clockwise = T, main = "Public remarks made by GOP Pr
# With Trump excluded, however, Republicans' proportion significantly decreased.
```

```
# Now we turn to analyze the words those speakers used alongside their mentioning of socialism. We need

wordcount <- function(speakers){
```

```
# This step gives texts' paragraphs in which speakers directly mentioned "socialism".
  address <- new_potus %>%
  filter(speaker == speakers) %>%
  select(text) %>%
  unlist() %>%
  str_split("\n")
address <- unlist(address)
on_soc <- address[str_detect(address, "socialis")]

# This step produces wordclouds for those paragraphs related to "socialism".
on_soc <- Corpus(VectorSource(on_soc))
on_soc <- DocumentTermMatrix(on_soc,
                             control = list(stopwords = TRUE,
                           tolower = TRUE,
                           removeNumbers = TRUE,
                           removePunctuation = TRUE))
freq <- colSums(as.matrix(on_soc))
sums <- as.data.frame(freq)
sums <- rownames_to_column(sums)
colnames(sums) <- c("term", "count")
sums <- arrange(sums, desc(count))
head <- sums[1:75,]
fin <- wordcloud::wordcloud(words = head$term, freq = head$count,
                      min.freq = 1000,
  max.words=100, random.order=FALSE, rot.per = 0.3)

return(fin)}

# We will first examine Trump's wordcloud.
wordcount("Donald J. Trump")

# Entering Trump's name and we found that "democrats" has been mentioned a lot alongside his comment on
# Trump also mentioned "communism" when talking about socialism. This is also a common rhetoric among c
# Another interesting note is the presence of word "booing" as a description of audience's reaction tow

wordcount("Barack Obama")

# Entering Obama's name and we found that "tax", "security", "government" are very frequently mentioned

# We also compare the wordclouds with those of Reagan and Carter. Noticeably, Carter also mentioned qui
wordcount("Ronald Reagan")
wordcount("Jimmy Carter")
```

Future Work

As I have said in the beginning, this analysis is generally brief. Due to the technical limits this analysis does not provide a detailed analysis of the contexts of politicians' reference to socialism. In general, though, socialism is still a tabooed word for mainstream politicians and most of them deny any associations with the concepts and practices related to that. I hope in the future there could be more precise text analysis on politicians' ideology patterns across time.