# Advanced Quantitative Methods in the Study of Political Behavior

## Web-scraping

## Today

- Course evaluations
- APIs and web scraping
- An extensive exercise

## Application Programming Interface (API)

- APIs are designed by the producers of data
  - **Twitter**: "Twitter data is the most comprehensive source of live, public conversation worldwide. Our REST, streaming, and Enterprise APIs enable programmatic analysis of Tweets back to the first Tweet in 2006."
  - **Facebook**: [Increasing restricted unfortunately]
  - **Wikimedia**: "The Wikimedia REST API offers access to Wikimedia's content and metadata in machine-readable formats."

## Application Programming Interface (API)

- APIs are designed by the producers of data
  - Twitter: `https://developer.twitter.com/`
  - Facebook:
    `https://www.facebook.com/ads/library/api/` [Old]
  - Wikimedia:
    `https://www.mediawiki.org/wiki/API:Main_page`

## Twitter as an example:

- Requires a Twitter "developer" account
  - Simple to get once you have a Twitter account
  - `https://developer.twitter.com/`
- Then need to create an "app" to collect data
- Twitter does not always easily provide access
  - App name
  - Application description
  - Website URL
  - "Tell us how this app will be used"

## Benefits:

- Twitter documentation is *excellent*
- Can download large amounts of social media data to better understand political behavior online
- Collecting data is generally free

## Drawbacks:

- Twitter can shut down your data collection application arbitrarily
- Not all data are free, and some types of data are prohibitively expensive
    - Much of it is still excellent for research, however
- Data can get big very fast
- Data require some extra programming effort to get the data into shape for analysis...

# Example tweet

```
{'created_at': 'Wed Nov 27 14:45:18 +0000 2019',
 'id': 1199700732804554752,
 'id_str': '1199700732804554752',
 'text': 'GOD BLESS THE U.S.A.! #MAGA\nhttps://t.co/CYkQGHAgcx',
 'truncated': False,
 'entities': {'hashtags': [{'text': 'MAGA', 'indices': [22, 27]}],
  'symbols': [],
  'user_mentions': [],
  'urls': [],
  'media': [{'id': 1199497812985204736,
    'id_str': '1199497812985204736',
    'indices': [28, 51],
    'media_url': 'http://pbs.twimg.com/ext_tw_video_thumb/1199497812985204736/pu/img/Mr8aWqotmFgatVGX.jpg
    'media_url_https': 'https://pbs.twimg.com/ext_tw_video_thumb/1199497812985204736/pu/img/Mr8aWqotmFgat
    'url': 'https://t.co/CYkQGHAgcx',
    'display_url': 'pic.twitter.com/CYkQGHAgcx',
    'expanded_url': 'https://twitter.com/DanScavino/status/1199498481905414144/video/1',
    'type': 'photo',
    'sizes': {'thumb': {'w': 150, 'h': 150, 'resize': 'crop'},
     'medium': {'w': 1200, 'h': 675, 'resize': 'fit'},
     'small': {'w': 680, 'h': 383, 'resize': 'fit'},
     'large': {'w': 1280, 'h': 720, 'resize': 'fit'}},
    'source_status_id': 1199498481905414144,
    'source_status_id_str': '1199498481905414144',
    'source_user_id': 620571475,
    'source_user_id_str': '620571475',
    'features': {}}]},
```

```
'extended_entities': {'media': [{'id': 1199497812985204736,
    'id_str': '1199497812985204736',
    'indices': [28, 51],
    'media_url': 'http://pbs.twimg.com/ext_tw_video_thumb/1199497812985204736/pu/img/Mr8aWqotmFgatVGX.jpg
    'media_url_https': 'https://pbs.twimg.com/ext_tw_video_thumb/1199497812985204736/pu/img/Mr8aWqotmFgatV
    'url': 'https://t.co/CYkQGHAgcx',
    'display_url': 'pic.twitter.com/CYkQGHAgcx',
    'expanded_url': 'https://twitter.com/DanScavino/status/1199498481905414144/video/1',
    'type': 'video',
    'sizes': {'thumb': {'w': 150, 'h': 150, 'resize': 'crop'},
     'medium': {'w': 1200, 'h': 675, 'resize': 'fit'},
     'small': {'w': 680, 'h': 383, 'resize': 'fit'},
     'large': {'w': 1280, 'h': 720, 'resize': 'fit'}},
    'source_status_id': 1199498481905414144,
    'source_status_id_str': '1199498481905414144',
    'source_user_id': 620571475,
    'source_user_id_str': '620571475',
    'video_info': {'aspect_ratio': [16, 9],
     'duration_millis': 73367,
     'variants': [{'content_type': 'application/x-mpegURL',
       'url': 'https://video.twimg.com/ext_tw_video/1199497812985204736/pu/pl/tGOZJACPDAo_rPnj.m3u8?tag=1
      {'bitrate': 2176000,
       'content_type': 'video/mp4',
       'url': 'https://video.twimg.com/ext_tw_video/1199497812985204736/pu/vid/1280x720/sJsjk2rI_oUyOPgu.m
      {'bitrate': 256000,
       'content_type': 'video/mp4',
       'url': 'https://video.twimg.com/ext_tw_video/1199497812985204736/pu/vid/480x270/y82THXLSFX7YT3ma.mp
      {'bitrate': 832000,
       'content_type': 'video/mp4',
       'url': 'https://video.twimg.com/ext_tw_video/1199497812985204736/pu/vid/640x360/cCPGN9HZOWBbgkyf.mp
    'features': {},
```

```
'additional_media_info': {'monetizable': False,
 'source_user': {'id': 620571475,
  'id_str': '620571475',
  'name': 'Dan Scavino',
  'screen_name': 'DanScavino',
  'location': 'Washington, DC New York',
  'description': 'MAKING AMERICA GREAT AGAIN! Personal Twitter Handle.',
  'url': None,
  'entities': {'description': {'urls': []}},
  'protected': False,
  'followers_count': 593038,
  'friends_count': 781,
  'listed_count': 3639,
  'created_at': 'Thu Jun 28 02:45:39 +0000 2012',
  'favourites_count': 2940,
  'utc_offset': None,
  'time_zone': None,
  'geo_enabled': True,
  'verified': True,
  'statuses_count': 10648,
  'lang': None,
  'contributors_enabled': False,
  'is_translator': False,
  'is_translation_enabled': False,
  'profile_background_color': '131516',
  'profile_background_image_url': 'http://abs.twimg.com/themes/theme1/bg.png',
  'profile_background_image_url_https': 'https://abs.twimg.com/themes/theme1/bg.png',
  'profile_background_tile': True,
  'profile_image_url': 'http://pbs.twimg.com/profile_1113590866009042944/J7GwR0cR_normal.jpg',
  'profile_image_url_https': 'https://pbs.twimg.com/profile_1113590866009042944/J7GwR0cR_normal.jpg',
  'profile_banner_url': 'https://pbs.twimg.com/profile_banners/620571475/1571389614',
```

```
    'profile_link_color': '1B95E0',
    'profile_sidebar_border_color': '000000',
    'profile_sidebar_fill_color': '000000',
    'profile_text_color': '000000',
    'profile_use_background_image': True,
    'has_extended_profile': False,
    'default_profile': False,
    'default_profile_image': False,
    'can_media_tag': False,
    'followed_by': False,
    'following': False,
    'follow_request_sent': False,
    'notifications': False,
    'translator_type': 'none'}}]},
'source': '<a href="http://twitter.com/download/iphone" rel="nofollow">Twitter for iPhone</a>',
'in_reply_to_status_id': None,
'in_reply_to_status_id_str': None,
'in_reply_to_user_id': None,
'in_reply_to_user_id_str': None,
'in_reply_to_screen_name': None,
'user': {'id': 25073877,
 'id_str': '25073877',
 'name': 'Donald J. Trump',
 'screen_name': 'realDonaldTrump',
 'location': 'Washington, DC',
 'description': '45th President of the United States of America',
 'url': 'https://t.co/OMxB0x7xC5',
 'entities': {'url': {'urls': [{'url': 'https://t.co/OMxB0x7xC5',
    'expanded_url': 'http://www.Instagram.com/realDonaldTrump',
    'display_url': 'Instagram.com/realDonaldTrump',
    'indices': [0, 23]}]},
  'description': {'urls': []}}
```

```
'protected': False,
'followers_count': 67057463,
'friends_count': 47,
'listed_count': 110919,
'created_at': 'Wed Mar 18 13:46:38 +0000 2009',
'favourites_count': 7,
'utc_offset': None,
'time_zone': None,
'geo_enabled': True,
'verified': True,
'statuses_count': 46489,
'lang': None,
'contributors_enabled': False,
'is_translator': False,
'is_translation_enabled': True,
'profile_background_color': '6D5C18',
'profile_background_image_url': 'http://abs.twimg.com/themes/theme1/bg.png',
'profile_background_image_url_https': 'https://abs.twimg.com/themes/theme1/bg.png',
'profile_background_tile': True,
'profile_image_url': 'http://pbs.twimg.com/profile_874276197357596672/kUuht00m_normal.jpg',
'profile_image_url_https': 'https://pbs.twimg.com/profile_874276197357596672/kUuht00m_normal.jpg',
'profile_banner_url': 'https://pbs.twimg.com/profile_banners/25073877/1560920145',
'profile_link_color': '1B95E0',
'profile_sidebar_border_color': 'BDDCAD',
'profile_sidebar_fill_color': 'C5CEC0',
'profile_text_color': '333333',
'profile_use_background_image': True,
```

```
 'has_extended_profile': False,
 'default_profile': False,
 'default_profile_image': False,
 'can_media_tag': False,
 'followed_by': False,
 'following': False,
 'follow_request_sent': False,
 'notifications': False,
 'translator_type': 'regular'},
'geo': None,
'coordinates': None,
'place': None,
'contributors': None,
'is_quote_status': False,
'retweet_count': 6286,
'favorite_count': 22944,
'favorited': False,
'retweeted': False,
'possibly_sensitive': False,
'lang': 'en'}
```

## Accessing the Twitter API:

- In Python: `tweepy` (& other libraries)
- In R: `rtweet` (& other libraries)

Both of these will ask for authorization keys that you will be given once Twitter approves your Twitter application

# Really cool research uses this type of data:

## How Censorship in China Allows Government Criticism but Silences Collective Expression

GARY KING *Harvard University*
JENNIFER PAN *Harvard University*
MARGARET E. ROBERTS *Harvard University*

*We offer the first large scale, multiple source analysis of the outcome of what may be the most extensive effort to selectively censor human expression ever implemented. To do this, we have devised a system to locate, download, and analyze the content of millions of social media posts originating from nearly 1,400 different social media services all over China before the Chinese government is able to find, evaluate, and censor (i.e., remove from the Internet) the subset they deem objectionable. Using modern computer-assisted text analytic methods that we adapt to and validate in the Chinese language, we compare the substantive content of posts censored to those not censored over time in each of 85 topic areas. Contrary to previous understandings, posts with negative, even vitriolic, criticism of the state, its leaders, and its policies are not more likely to be censored. Instead, we show that the censorship program is aimed at curtailing collective action by silencing comments that represent, reinforce, or spur social mobilization, regardless of content. Censorship is oriented toward attempting to forestall collective activities that are occurring now or may occur in the future—and, as such, seem to clearly expose government intent.*

**The Chinese government censors posts concerning collective action (i.e. potential protest), not those critical of the government**

# From Isolation to Radicalization: Anti-Muslim Hostility and Support for ISIS in the West

TAMAR MITTS *Columbia University*

*What explains online radicalization and support for ISIS in the West? Over the past few years, thousands of individuals have radicalized by consuming extremist content online, many of whom eventually traveled overseas to join the Islamic State. This study examines whether anti-Muslim hostility might drive pro-ISIS radicalization in Western Europe. Using new geo-referenced data on the online behavior of thousands of Islamic State sympathizers in France, the United Kingdom, Germany, and Belgium, I study whether the intensity of anti-Muslim hostility at the local level is linked to pro-ISIS radicalization on Twitter. The results show that local-level measures of anti-Muslim animosity correlate significantly and substantively with indicators of online radicalization, including posting tweets sympathizing with ISIS, describing life in ISIS-controlled territories, and discussing foreign fighters. High-frequency data surrounding events that stir support for ISIS—terrorist attacks, propaganda releases, and anti-Muslim protests—show the same pattern.*

**Offline anti-Muslim animosity is linked to an increase in online radicalization**

POLITICAL SCIENCE

# Fake news on Twitter during the 2016 U.S. presidential election

**Nir Grinberg[1,2]*, Kenneth Joseph[3]*, Lisa Friedland[1]*, Briony Swire-Thompson[1,2], David Lazer[1,2]†**

The spread of fake news on social media became a public concern in the United States after the 2016 presidential election. We examined exposure to and sharing of fake news by registered voters on Twitter and found that engagement with fake news sources was extremely concentrated. Only 1% of individuals accounted for 80% of fake news source exposures, and 0.1% accounted for nearly 80% of fake news sources shared. Individuals most likely to engage with fake news sources were conservative leaning, older, and highly engaged with political news. A cluster of fake news sources shared overlapping audiences on the extreme right, but for people across the political spectrum, most political news exposure still came from mainstream media outlets.

**Fake news is actually very rare, and shared mostly by old, highly politically interested conservatives**

# A 61-million-person experiment in social influence and political mobilization

Robert M. Bond[1], Christopher J. Fariss[1], Jason J. Jones[2], Adam D. I. Kramer[3], Cameron Marlow[3], Jaime E. Settle[1] & James H. Fowler[1,4]

**Voting is contagious, and affected by social signals from friends, and friends of friends**

# Exposure to opposing views on social media can increase political polarization

**Christopher A. Bail[a,1], Lisa P. Argyle[b], Taylor W. Brown[a], John P. Bumpus[a], Haohan Chen[c], M. B. Fallin Hunzaker[d], Jaemin Lee[a], Marcus Mann[a], Friedolin Merhout[a], and Alexander Volfovsky[e]**

[a]Department of Sociology, Duke University, Durham, NC 27708; [b]Department of Political Science, Brigham Young University, Provo, UT 84602; [c]Department of Political Science, Duke University, Durham, NC 27708; [d]Department of Sociology, New York University, New York, NY 10012; and [e]Department of Statistical Science, Duke University, Durham, NC 27708

**Exposure to opposing political viewpoints increases polarization**

POLITICAL SCIENCE

# Exposure to ideologically diverse news and opinion on Facebook

Eytan Bakshy,[1*†] Solomon Messing,[1†] Lada A. Adamic[1,2]

Exposure to news, opinion, and civic information increasingly occurs through social media. How do these online networks influence exposure to perspectives that cut across ideological lines? Using deidentified data, we examined how 10.1 million U.S. Facebook users interact with socially shared news. We directly measured ideological homophily in friend networks and examined the extent to which heterogeneous friends could potentially expose individuals to cross-cutting content. We then quantified the extent to which individuals encounter comparatively more or less diverse content while interacting via Facebook's algorithmically ranked News Feed and further studied users' choices to click through to ideologically discordant content. Compared with algorithmic ranking, individuals' choices played a stronger role in limiting exposure to cross-cutting content.

**News feed algorithms do not strongly affect the ideological diversity of what we read online**

## Unfortunately, not all data can be accessed by an API

- Data on many websites are not neatly packaged
- Sometimes the data of interest is the website itself
- No straightforward way to collect data from websites that don't have an API

## Solution is to automate data collection through "scraping"

- An ad hoc procedure because every website is different
- Often data collection is imperfect
- Requires post-collection data cleaning

## APIs versus web scraping

- Application Programming Interface (API)
  - Formal protocols that allow you to access a product or service online
- Web scraping
  - Extract information from websites without access to any formal mechanism designed for that purpose

## Options

- In R: `rvest` (by Hadley Wickham)
- In Python: `BeautifulSoup` & `Selenium`
- These the main ones, but there are other options

**Today we will use `rvest` to scrape:**

- Danish municipality vote shares results
- Text from Elizabeth Warren's plans on her 2020 election campaign website

# Exercise

# Exercise solution

```
# Scrape the individual plan URLs
main_site_url <- "https://elizabethwarren.com/plans"
plan_selector <- ".jOIKMx"

# Read in the data from the main page of Warren's plans website
main_site_url_html <- read_html(main_site_url)

# Pull out the URLs for each of the plans
plan_urls <- main_site_url_html %>%
             html_nodes(css = plan_selector) %>% # html_attrs()
             html_attr("href")

# Clean plan_urls so we just have URLs that point to plans
# i.e. each entry should look something like:
# "https://elizabethwarren.com/plans/ultra-millionaire-tax"
# You will have to do a bit of cleaning of the plan_urls
# here to get just the URLs for the 60 plans
plan_urls <- as.character(na.omit(plan_urls))
plan_urls <- paste0("https://elizabethwarren.com", plan_urls)
```

```r
# Pull the raw data from each plan
# ***** PLEASE GRAB ONLY 5 PLANS TO START WITH
# ***** WE DON'T WANT TO HAMMER OUR FRIEND ELIZABETH'S SITE
# Once _all_ of your code is working, then you can run it once again to get
# all of the data for all 60 plans
raw_url_data <- list()

# Loop through each of the plan URLs and grab the html
# Use length(plan_urls) instead of 5 to get them all, but only once you have
# all of the code in this file working, or else we'll end up hammering Warren's
# site and get IP blocked/banned
for(i in 1:length(plan_urls)) {

  cat(length(plan_urls) - i + 1, "")
  flush.console()
  raw_url_data[[i]] <- read_html(plan_urls[i])
  Sys.sleep(2) # Sleep for 2 seconds to prevent hitting the site too fast

}
```

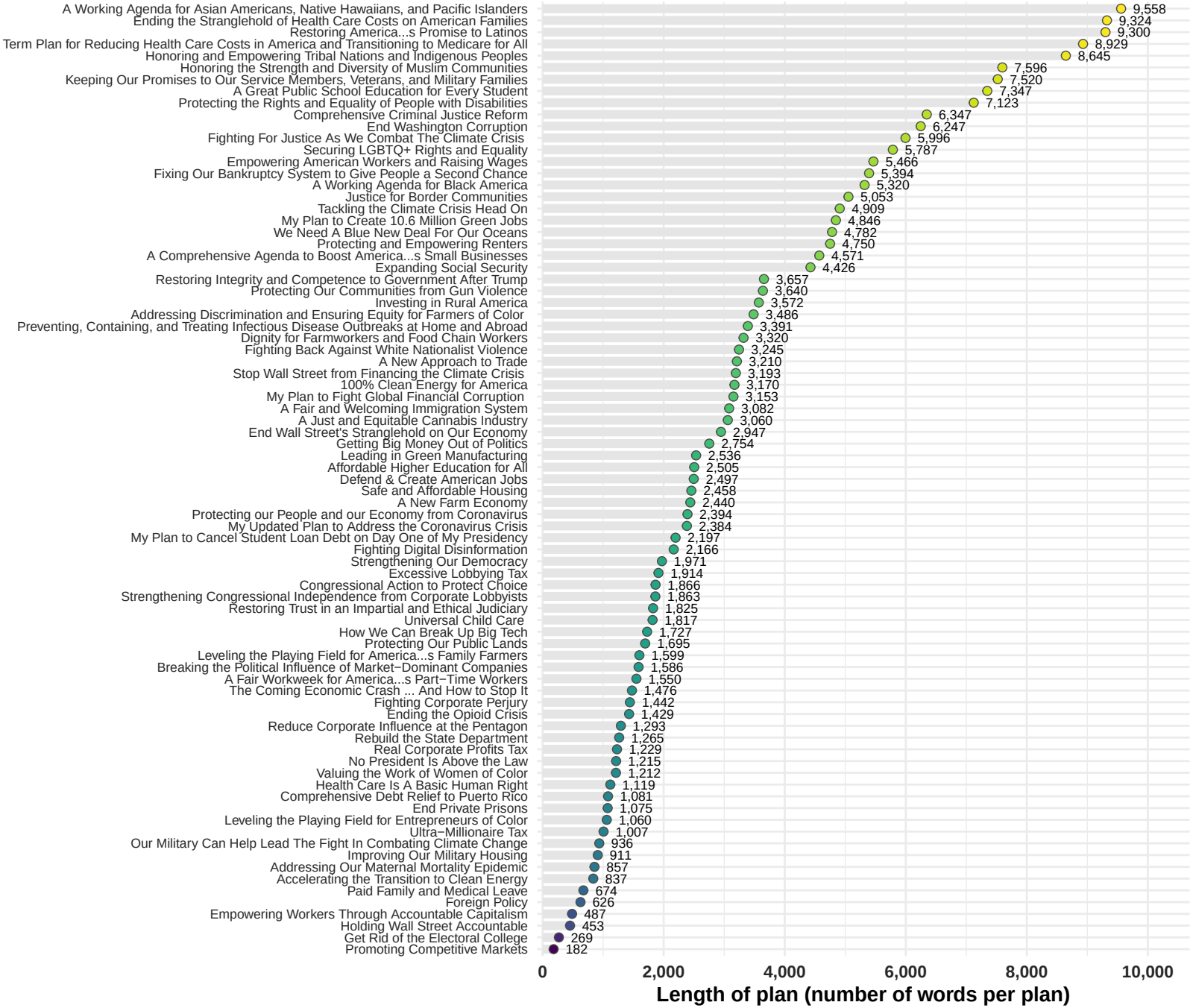
```r
# Pull out the text we want from each of Elizabeth Warren's plans
# Title of each plan
title_selector <- ".mCEuH"
# The full text from each plan
main_text_selector <- ".ZMNLV"

# This creates an empty data.frame where the resulting data will be stored
# The [-1, ] just remove the first and only row of the data.frame so that
# there are no data whatsoever in the data.frame
D <- data.frame(title = "", main_text = "", num_words = 0)[-1, ]
```

```r
for(i in 1:length(raw_url_data)) {

  cat(length(raw_url_data) - i + 1, "")

  # Plan title
  title <- raw_url_data[[i]] %>%
           html_nodes(css = title_selector) %>%
           html_text()

  # Text of the plan
  main_text <- raw_url_data[[i]] %>%
               html_nodes(css = main_text_selector) %>%
               html_text()

  # This takes the vector of text and makes a single string out of it
  # i.e. from c("Here is my plan.", "What my plan argues for is...")
  #        to c("Here is my plan. What my plan argues for is...")
  main_text <- paste(main_text, collapse = " ")

  # Number of words in the plan
  num_words <- str_count(main_text, pattern = " ")

  # Store data
  Plan <- data.frame(title = title,
                     main_text = main_text,
                     num_words = num_words)

  # Add new place data to the final dataset
  D <- rbind(D, Plan)

}
```

```
# Graph the number of words per plan as a dot plot i.e. geom_point()
# y axis: title of the plan. x axis: number of words in the plan.
# Order the graph such that the plan with the most words is at the top
# and the plan with the least words is at the bottom
pdf("~/Downloads/Warren.pdf", 9.5, 8.25, useDingbats = FALSE)
ggplot(D, aes(x = num_words, y = fct_reorder(title, num_words),
              fill = log(num_words))) +
  labs(title = "How Long are Elizabeth Warren's Plans?",
       subtitle = "The Number of Words in Warren's 60 Plans",
       caption  = "Source: https://elizabethwarren.com/plans",
       x = "Length of plan (number of words per plan)", y = "") +
  coord_cartesian(xlim = c(400, 10200)) +
  scale_x_continuous(breaks = seq(0, 10000, by = 2000),
                     labels = formatC(seq(0, 10000, 2000),
                     format="f", big.mark=",", digits = 0)) +
  geom_segment(aes(yend = fct_reorder(title, num_words), xend = 0),
               size = 2.25, color = "grey90") +
  geom_point(color = "grey30", shape = 21, size = 2, stroke = 0.4) +
  geom_text(aes(label = formatC(num_words, format="f", big.mark=",", digits = 0)),
                hjust = 0, nudge_x = 200, size = 2.4) +
  scale_fill_viridis(option = "viridis") +
  theme_minimal() +
  theme(legend.position = "none",
        axis.text.x = element_text(colour = "grey15", face = "bold"),
        axis.text.y = element_text(size = 6.5, colour = "grey15"),
        axis.title.x = element_text(face = "bold"),
        plot.title = element_text(hjust = 0, face = "bold"),
        plot.subtitle = element_text(hjust = 0),
        plot.title.position = "plot",
        plot.caption.position =  "plot")
dev.off()
```

# How Long are Elizabeth Warren's Plans?

The Number of Words in Warren's 60 Plans




Source: https://elizabethwarren.com/plans