Final Project Web Scraping

RcCatedral

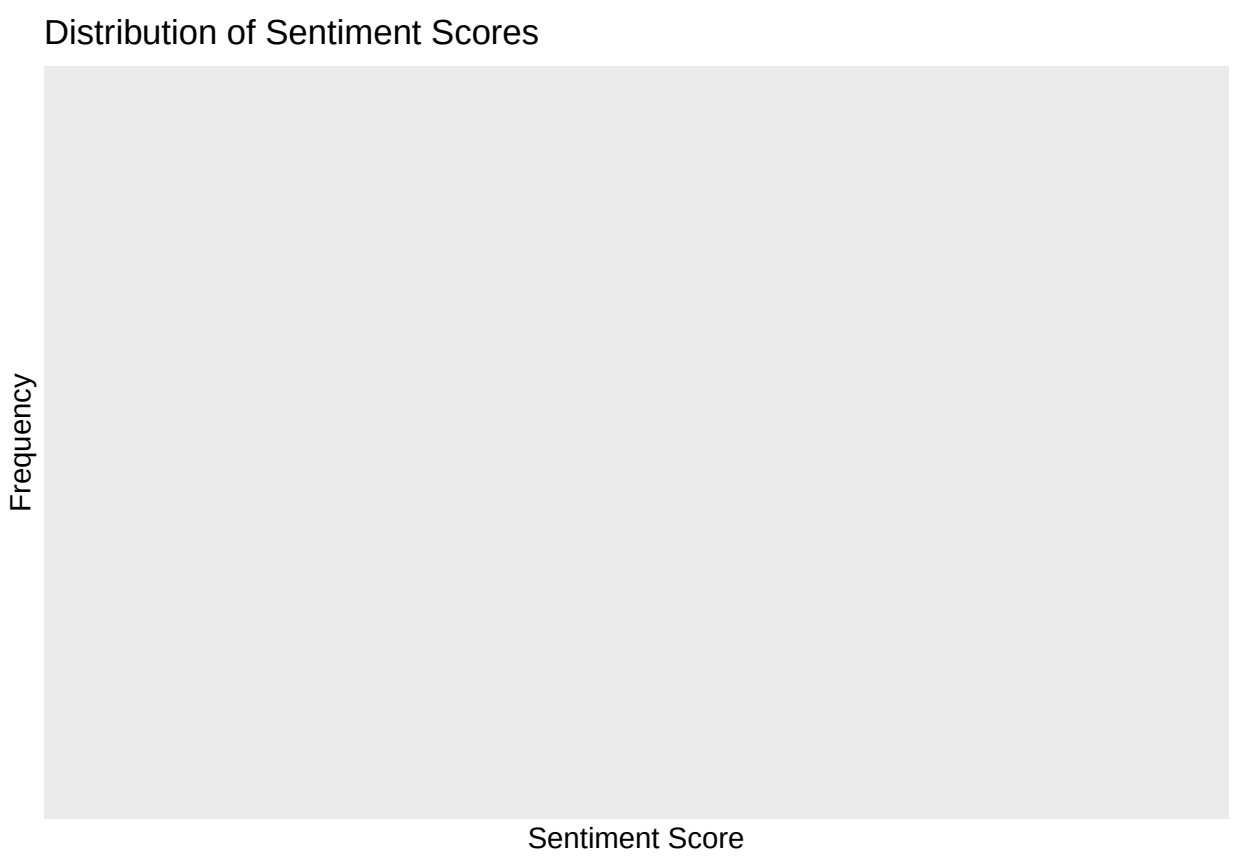
2023-12-22

```
library(rvest)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.4.4     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter()         masks stats::filter()
## x readr::guess_encoding() masks rvest::guess_encoding()
## x dplyr::lag()            masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(tidytext)

url <- "https://www.amazon.com/Redragon-Keyboard-Mechanical-Software-Supported/dp/B09BVCVTBC/ref=sr_1_2

data_scrape <- read_html(url)

user_name <- data_scrape %>%
  html_nodes(".a-profile-name") %>%
  html_text()

keyboard_rating <- data_scrape %>%
  html_nodes(".review-rating") %>%
  html_text()

reviews <- data_scrape %>%
  html_nodes(".review-text-content span") %>%
  html_text()

max_length <- max(length(user_name), length(keyboard_rating), length(reviews))
user_name <- rep(user_name, length.out = max_length)
keyboard_rating <- rep(keyboard_rating, length.out = max_length)
reviews <- rep(reviews, length.out = max_length)

analysis_data <- data.frame(user_name, keyboard_rating, reviews)

analysis_data <- analysis_data %>%
  unnest_tokens(word, reviews) %>%
  inner_join(get_sentiments("afinn"), by = "word") %>%
```

```
  group_by(user_name) %>%
  summarize(sentiment_score = sum(value, na.rm = TRUE))

ggplot(analysis_data, aes(x = sentiment_score)) +
  geom_histogram(binwidth = 1, fill = "red", color = "orange") +
  labs(title = "Distribution of Sentiment Scores", x = "Sentiment Score", y = "Frequency")
```




```
top_users <- analysis_data %>%
  arrange(sentiment_score) %>%
  slice_head(n = 5) %>%
  bind_rows(analysis_data %>%
              arrange(desc(sentiment_score)) %>%
              slice_tail(n = 5))

ggplot(top_users, aes(x = reorder(user_name, sentiment_score), y = sentiment_score, fill = user_name)) +
  geom_bar(stat = "identity") +
  coord_flip() +
  labs(title = "Users with Highest and Lowest Sentiment Scores", x = "User", y = "Sentiment Score")
```

Users with Highest and Lowest Sentiment Scores