Topic 04 - Sentiment Analysis II

Julia Parish

2022-04-26

Sentiment Analyis II

This text sentiment analysis was completed as an assignment for the course, Environmental Data Science 231: Text and Sentiment Analysis for Environmental Problems. The data was sourced from Twitter.

Original assignment instructions can be found here

Load Libraries

```
library(quanteda)
library(quanteda.sentiment)
library(quanteda.textstats)
library(tidyverse)
library(tidytext)
library(lubridate)
library(wordcloud)
library(reshape2)
library(here)
library(rtweet)
library(paletteer)
```

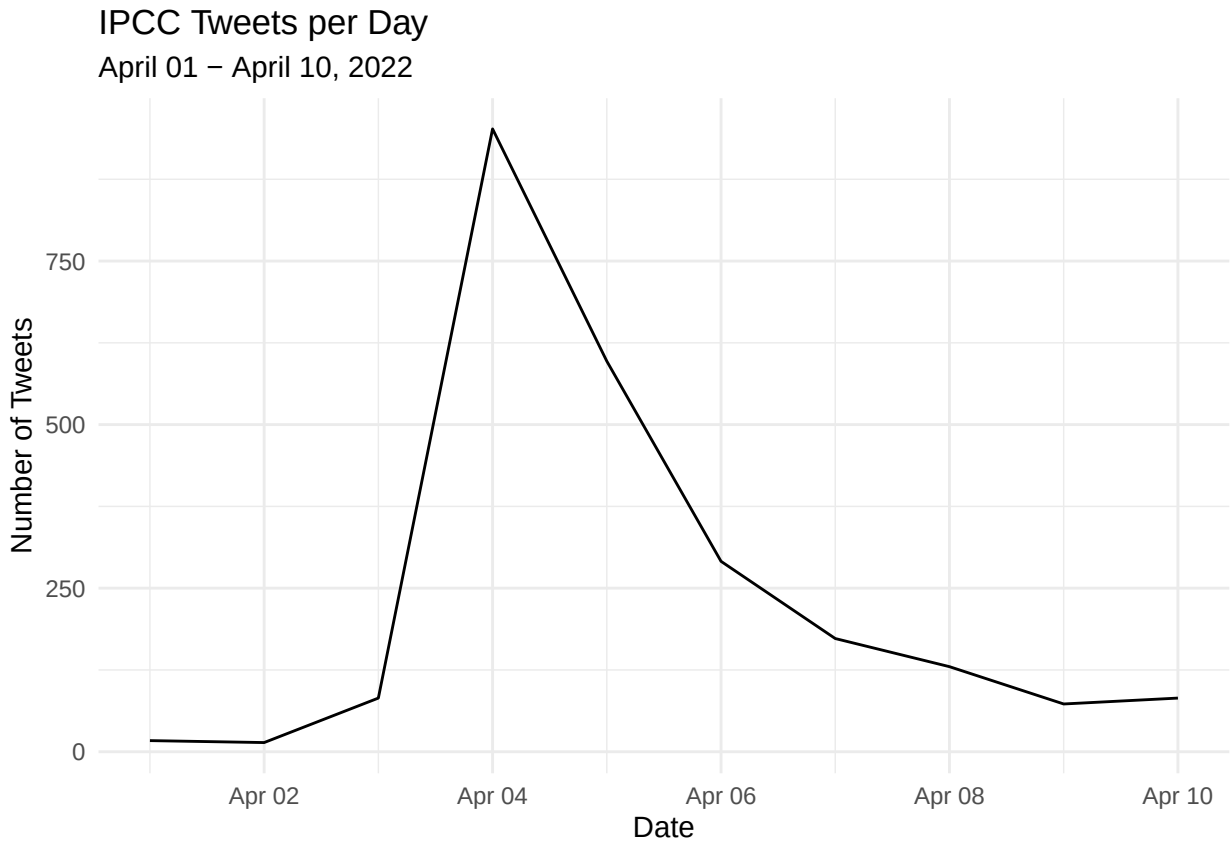
Load IPPC tweet data & create plot of data

```
raw_tweets <- read.csv("https://raw.githubusercontent.com/MaRo406/EDS_231-text-sentiment/main/dat/IPCC_

dat<- raw_tweets[,c(5,7)] # Extract Date and Title fields

tweets <- tibble(text = dat$Title,
                 id = seq(1:length(dat$Title)),
                 date = as.Date(dat$Date,'%m/%d/%y'))

head(tweets$text, n = 10)
```

```
##  [1] "thank you, followers, for the great photo suggestions for our upcoming IPCC report - on Monday
##  [2] "Greenpeace: The real solution to the climate crisis will require a rapid transition away from
##  [3] "Governments have a responsibility to ensure that #IPCCReport is grounded in rapid phaseout of
not #FalseClimateSolutions. \n\nRead more in our open letter: https://t.co/4larBPgeba https://t.co/Fv1Op
##  [4] "Next week, the IPCC will publish a new report detailing their new models and policy pathways.
##  [5] "Live stream of virtual IPCC press conference releasing the report on mitigation of climate cha
##  [6] "Attention journalists: The deadline for embargoed materials for the upcoming @IPCC_CH report o
##  [7] "The IPCC Report and "The Physics of Climate Change" https://t.co/xnxP3fup2a"
##  [8] "With time running short and most of the Summary for Policymakers yet to be approved, #IPCC Wor
```

```
##  [9] "A helpful perspective on how to talk about the scenarios discussed in the forthcoming IPCC repo
## [10] "The private sector is an integral component of the water cycle and has much to lose as critical
```

```r
#simple plot of tweets per day
tweets %>%
  count(date) %>%
  ggplot(aes(x = date, y = n))+
  geom_line() +
  labs(title = "IPCC Tweets per Day",
       subtitle = "April 01 - April 10, 2022",
       x = "Date",
       y = "Number of Tweets") +
  theme_minimal()
```



Questions

1. Think about how to further clean a twitter data set. Let's assume that the mentions of twitter accounts is not useful to us. Remove them from the text field of the tweets tibble.

```r
# keep original text column to track changes
tweets_clean <- tweets %>%
  mutate(text_clean = text)

# remove mentions and website links
tweets_clean$text_clean <- str_remove(tweets_clean$text_clean, "@[a-z,A-Z]*")
```

```r
tweets_clean$text_clean <- str_remove(tweets_clean$text_clean, "[:digit:]")

tweets_clean$text_clean <- gsub("http.*","", tweets_clean$text_clean)

tweets_clean$text_clean <- gsub("https.*","", tweets_clean$text_clean)

# remove punctuations
tweets_clean$text_clean <- gsub('[[:punct:]]', '', tweets_clean$text_clean)

#tokenise tweets and remove stop words
words <- tweets_clean %>%
  select(id, date, text, text_clean) %>%
  unnest_tokens(output = word, input = text_clean, token = "words") %>%
  anti_join(stop_words, by = "word")

#clean tokens
# remove numbers
clean_tokens <- str_remove_all(words$word, "[:digit:]")

# remove mentions
clean_tokens <- str_remove_all(clean_tokens, "@[a-z,A-Z]*")

# remove apostrophes
clean_tokens <- gsub("'s", '', clean_tokens)

# remove unnecessary twitter formats
clean_tokens <- str_remove_all(clean_tokens, "t.co")

# stem the token "ipcc" as there are some plural instances
clean_tokens <- str_replace_all(string = clean_tokens,
                                pattern = "ipcc[a-z, A-Z]*",
                                replacement = "ipcc")


# stem the token "fuel" as it may occur in the plural form
clean_tokens <- str_replace_all(string = clean_tokens,
                                pattern = "fuel[a-z, A-Z]*",
                                replacement = "fuel")

# stem the token "biofuel" as it may occur in the plural form
clean_tokens <- str_replace_all(string = clean_tokens,
                                pattern = "biofuel[a-z, A-Z]*",
                                replacement = "biofuel")

# stem the token "headline" as it may occur in the plural form
clean_tokens <- str_replace_all(string = clean_tokens,
                                pattern = "headline[a-z, A-Z]*",
                                replacement = "headline")

# stem the token "regulation" as it may occur in the plural form
clean_tokens <- str_replace_all(string = clean_tokens,
                                pattern = "regulation[a-z, A-Z]*",
                                replacement = "regulation")
```

```r
# stem the token "follower" as it may occur in the plural form
clean_tokens <- str_replace_all(string = clean_tokens,
                                pattern = "follower[a-z, A-Z]*",
                                replacement = "follower")

# stem the token "suggestion" as it may occur in the plural form
clean_tokens <- str_replace_all(string = clean_tokens,
                                pattern = "suggestion[a-z, A-Z]*",
                                replacement = "suggestion")

# stem the token "solution" as it may occur in the plural form
clean_tokens <- str_replace_all(string = clean_tokens,
                                pattern = "solution[a-z, A-Z]*",
                                replacement = "solution")

# stem the token "reduction" as it may occur in the plural form
clean_tokens <- str_replace_all(string = clean_tokens,
                                pattern = "reduction[a-z, A-Z]*",
                                replacement = "reduction")

# stem the token "risk" as it may occur in the plural form
clean_tokens <- str_replace_all(string = clean_tokens,
                                pattern = "risk[a-z, A-Z]*",
                                replacement = "risk")

# stem the token "scenario" as it may occur in the plural form
clean_tokens <- str_replace_all(string = clean_tokens,
                                pattern = "scenario[a-z, A-Z]*",
                                replacement = "scenario")

# stem the token "submission" as it may occur in the plural form
clean_tokens <- str_replace_all(string = clean_tokens,
                                pattern = "submission[a-z, A-Z]*",
                                replacement = "submission")

words$clean <- clean_tokens

# remove the empty strings
tib <-subset(words, clean != "")

#reassign
words <- tib

head(words)
```

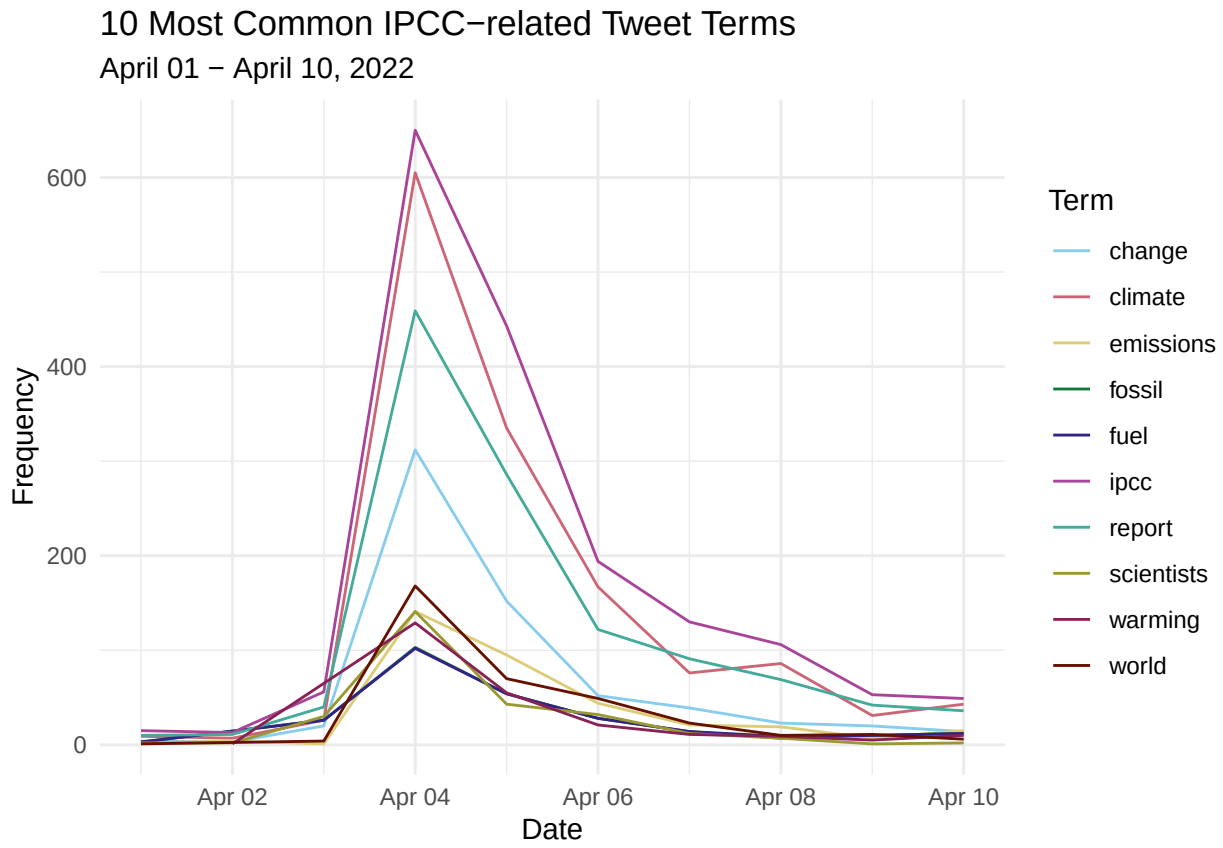
```
## # A tibble: 6 x 5
##      id date       text                                                word  clean
##   <int> <date>     <chr>                                               <chr> <chr>
## 1     1 2022-04-01 "thank you, followers, for the great photo sugge~ foll~ foll~
## 2     1 2022-04-01 "thank you, followers, for the great photo sugge~ photo photo
## 3     1 2022-04-01 "thank you, followers, for the great photo sugge~ sugg~ sugg~
## 4     1 2022-04-01 "thank you, followers, for the great photo sugge~ upco~ upco~
## 5     1 2022-04-01 "thank you, followers, for the great photo sugge~ ipcc  ipcc
```

```
## 6     1 2022-04-01 "thank you, followers, for the great photo sugge~ repo~ repo~
```

2. Compare the ten most common terms in the tweets per day. Do you notice anything interesting?

```
words_freq <- words %>%
  group_by(clean) %>%
  summarise(n()) %>%
  top_n(10) %>%
  rename("freq" = "n()") %>%
  select(clean)

words_top10 <- inner_join(words_freq, words, by = "clean") %>%
  group_by(date, clean) %>%
  summarize(n()) %>%
  rename("freq" = "n()")
```

```
top10term_plot <- ggplot(data = words_top10, aes(x = date, y = freq)) +
  geom_line(aes(color = clean)) +
  labs(title = "10 Most Common IPCC-related Tweet Terms",
       subtitle = "April 01 - April 10, 2022",
       x = "Date",
       y = "Frequency",
       color = "Term") +
  scale_color_paletteer_d("rcartocolor::Safe") +
  theme_minimal()

top10term_plot
```



3. Adjust the wordcloud in the "wordcloud" chunk by coloring the positive and negative words so they are identifiable.

```
#load sentiment lexicons
bing_sent <- get_sentiments('bing')
nrc_sent <- get_sentiments('nrc')
```

```
cloud <- words %>% inner_join(get_sentiments("bing")) %>%
  inner_join(get_sentiments("nrc")) %>%
  count(word, sentiment, sort = TRUE) %>%
  acast(word ~ sentiment, value.var = "n", fill = 0) %>%
  comparison.cloud(colors = c("slateblue3", "goldenrod2"),
                   max.words = 100)
```


```
cloud
```

```
## NULL
```

4. Let's say we are interested in the most prominent entities in the Twitter discussion. Which are the top 10 most tagged accounts in the data set. Hint: the "explore_hashtags" chunk is a good starting point.

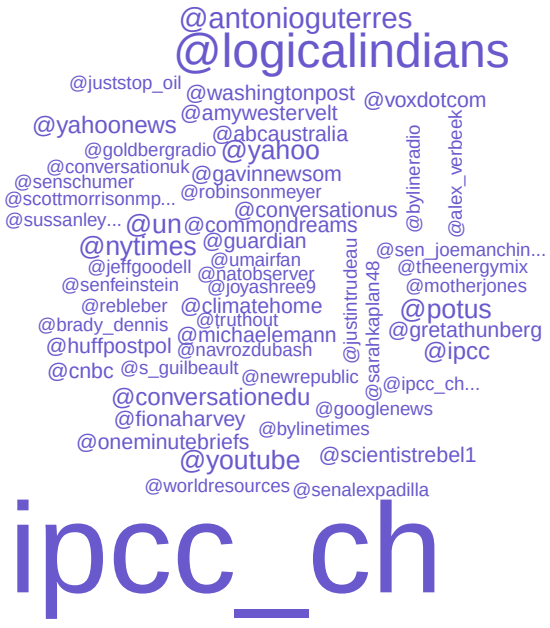
```
corpus <- corpus(dat$Title) #enter quanteda
#summary(corpus)
# text: tweet ID, Types: species words, Tokens: total words
```

```
tagged_accts <- tokens(corpus, remove_punct = TRUE) %>%
                tokens_keep(pattern = "@*")

# feature matrix - shows location of each features in the corpus aka located in the tweet : document fe
dfm_tags<- dfm(tagged_accts)

# frequency of hashtags
tstat_freq <- textstat_frequency(dfm_tags, n = 100)
head(tstat_freq, 10)
```

```
##             feature frequency rank docfreq group
## 1          @ipcc_ch       131    1     131   all
## 2   @logicalindians        38    2      38   all
## 3  @antonioguterres        16    3      16   all
## 4          @nytimes        14    4      14   all
## 5            @yahoo        14    4      14   all
```

```
## 6            @potus        13    6      13   all
## 7              @un         12    7      12   all
## 8         @youtube         11    8      11   all
## 9  @conversationedu         10    9      10   all
## 10           @ipcc          9   10       9   all
```

```
#tidytext gives us tools to convert to tidy from non-tidy formats
tags_tib <- tidy(dfm_tags)

tags_tib %>%
   count(term) %>%
   with(wordcloud(term, n, color = "slateblue3", max.words = 100))
```




```
top_tags <- tags_tib %>%
  group_by(term) %>%
  summarize(n()) %>%
  rename("freq" = "n()") %>%
  top_n(10)
```

```
top10user_plot <- top_tags %>%
  mutate(term = fct_relevel(term,
            "@ipcc_ch", "@logicalindians", "@antonioguterres", "@nytimes", "@yahoo", "@potus", "@un", "@
  ggplot(aes(x = freq, y = term)) +
  geom_point(color = "slateblue2") +
  labs(title = "Top 10 Tagged IPCC-related Accounts",
       subtitle = "April 01 - April 10, 2022",
       x = "Frequency of Mentions",
       y = "Twitter Account",
       caption = "Data Source: Twitter") +
  theme_minimal()

top10user_plot
```

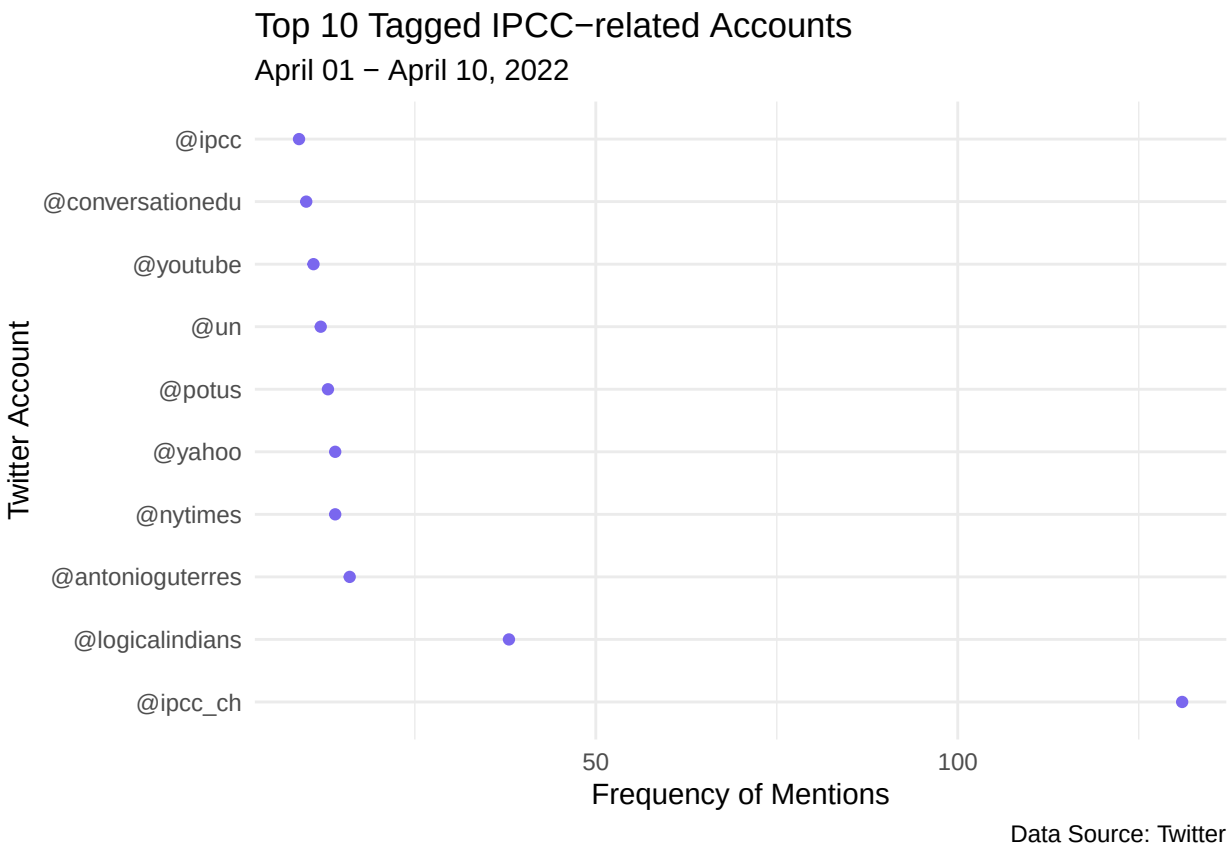


Figure 1: Top 10 Twitter Accounts tagged in IPCC related tweets between April 1 - April 10, 2022