Topic 3 - Sentiment Analysis I

Julia Parish

2022-04-19

Sentiment Analyis I

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 Nexis Uni. Original assignment instructions can be found here.

Load Packages

```
library(tidyr) #text analysis in R
library(lubridate) #working with date data
library(pdftools) #read in pdfs
library(tidyverse)
library(tidytext)
library(here)
library(LexisNexisTools) #Nexis Uni data wrangling
library(sentimentr)
library(readr)
library(here)
```

0. Using the "IPCC" Nexis Uni data set from the class presentation and the pseudo code discussed in class, recreate Figure 1A from Froelich et al. 2017 (Date x # of 1) positive, 2) negative, 3) neutral headlines):

```
data <- "/Users/julia/Documents/_MEDS/04_spring/EDS231_TextSentiment/repository/EDS231_TextSentimentAna

ipcc_data <- list.files(pattern = "Nexis_IPCC_Results.docx", path = data,
                        full.names = TRUE, recursive = TRUE, ignore.case = TRUE)

# lnt_read = read in a LexisNexis file
dat <- lnt_read(ipcc_data) #Object of class 'LNT output'
```

```
meta_df <- dat@meta
articles_df <- dat@articles
paragraphs_df <- dat@paragraphs
```

```
dat2<- data_frame(element_id = seq(1:length(meta_df$Headline)), Date = meta_df$Date,
                  Headline = meta_df$Headline)
```

```
paragraphs_dat <- data_frame(element_id = paragraphs_df$Art_ID, Text  = paragraphs_df$Paragraph)

dat3 <- inner_join(dat2, paragraphs_dat, by = "element_id")
```

```
headsen <- get_sentences(dat2$Headline)
sent <- sentiment(headsen)
sent_df <- inner_join(dat2, sent, by = "element_id")
sentiment <- sentiment_by(sent_df$Headline)
sent_df <- sent_df%>%
  arrange(sentiment)
```

```
sent_summary_df <- sent_df %>%
  mutate(sent_category = case_when(
    sentiment < 0 ~ "negative",
    sentiment > 0 ~ "positive",
    sentiment == 0 ~ "neutral")) %>%
  group_by(Date, sent_category) %>%
  summarise(num_headlines = n())
```

```
ggplot(data = sent_summary_df, aes(x = Date, y = num_headlines)) +
  geom_line(aes(color = sent_category), size = 1.5) +
  scale_color_manual(name = "",
                     values = c("red1", "gray53", "royalblue2"),
                     labels = c ("Negative", "Neutral", "Positve")) +
  labs(title = "Sentiment Analysis of IPCC Headlines",
       caption = "Data Source: Lexis Nexus",
       x = " ",
       y = "Media Sentiment\n(no. of headlines)") +
  theme_classic() +
  theme(legend.position = c(0.8, 0.8),
        legend.background = element_blank(),
        axis.text.x = element_text(angle = 25, hjust=1)) +
  scale_x_date(date_labels = "%b %d, %Y",
               limits = c(as.Date("2022-04-04"), as.Date("2022-04-11")),
               breaks = "1 day")
```

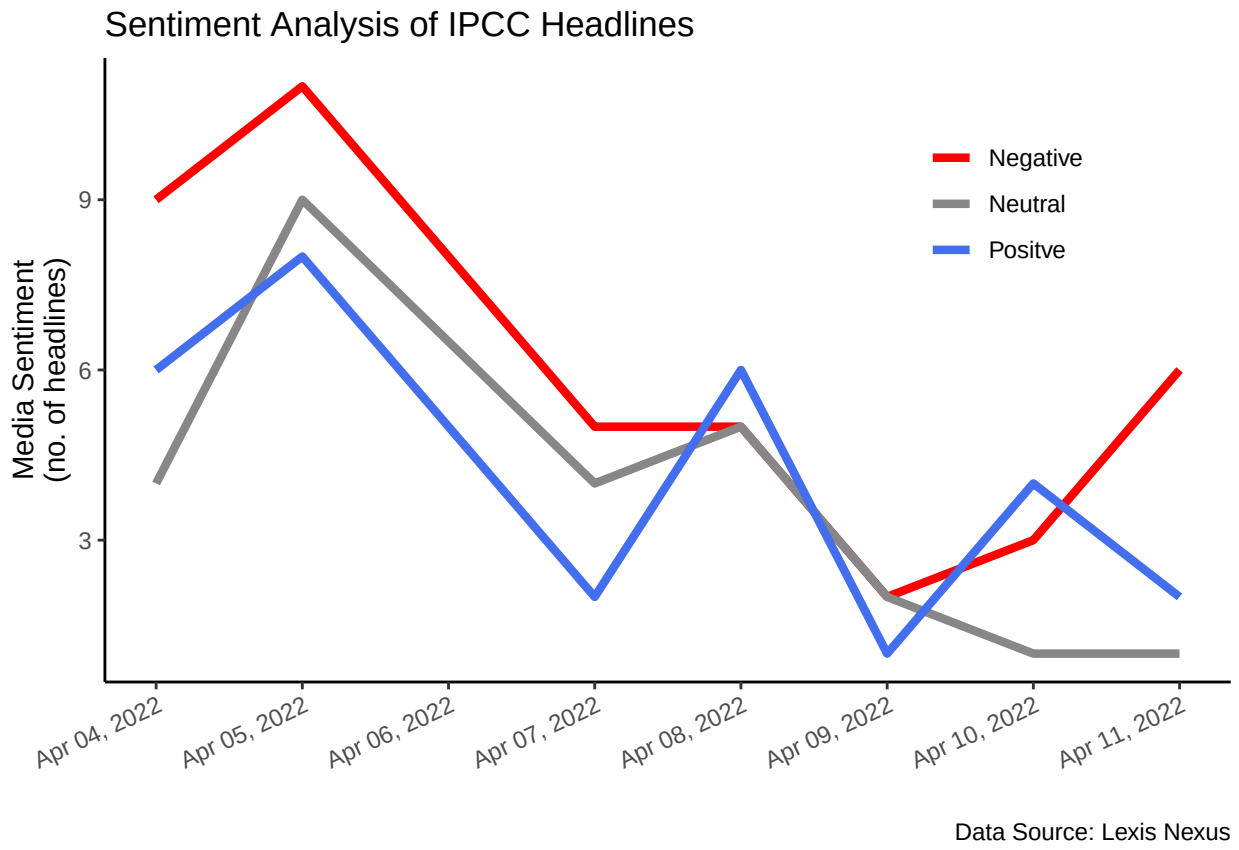

Figure 1: Newspaper 'IPCC' media sentiment. Sentiment over time based on the frequency of newspaper headlines with negative (red), positive (blue), and neutral (gray) titles.

1. Access the Nexis Uni database through the UCSB library:

I navigated to the UC Santa Barbara Library Nexis Uni portal.

2. Choose a key search term or terms to define a set of articles.

The search term I chose is "invasive species". I used the 'Guided Search' function and made the following search parameter selections:

- Publication type: 'News'
- Search term: 'invasive species'
- Date range: April 17, 2021 thru April 17, 2022.

3. Use your search term along with appropriate filters to obtain and download a batch of at least 100 full text search results (.docx).

Saved search results as file in my repo data folder: 'invasivespecies.docx'.

4. Read your Nexis article document into RStudio.

```
invspe <- list.files(pattern = "invasivespecies.docx", path = data, full.names = TRUE,
                     recursive = TRUE, ignore.case = TRUE)

dat_invspe <- lnt_read(invspe)
```

5. This time use the full text of the articles for the analysis. First clean any artifacts of the data collection process (hint: this type of thing should be removed: "Apr 04, 2022(Biofuels Digest: http://www.biofuelsdigest.com/ Delivered by Newstex"))

```
# create variable for headlines, remove duplicates
meta_df <- dat_invspe@meta %>% # 1st tibble in dat
  distinct(Headline, .keep_all = TRUE)

# create variable for articles, remove duplicates
articles_df <- dat_invspe@articles %>%
  distinct(Article, .keep_all = TRUE)

# create variable for paragraphs, remove duplicates
paragraphs_df <- dat_invspe@paragraphs %>%
  distinct(Paragraph, .keep_all = TRUE)
```

```
# articles
dat4 <- tibble(element_id = seq(1:length(meta_df$Headline)),
                   Date = meta_df$Date,
                   Headline = meta_df$Headline)

paragraphs_dat <- data_frame(element_id = paragraphs_df$Art_ID,
                             Text  = paragraphs_df$Paragraph)
```

```r
# join dat4 with paragraphs_df
dat5 <- inner_join(dat4, paragraphs_dat, by = "element_id")


# remove paragraphs that contain website links and are shorter than 41 characters
dat6 <- dat5 %>%
  mutate(link = str_detect(dat5$Text, "1]: http://", negate = FALSE),
         txt_short = nchar(dat5$Text) < 40) %>%
  filter(link == FALSE & txt_short == FALSE) %>%
  select(!c(link, txt_short))

# Summary of rows removed
rows_all = nrow(dat5)
rows_new = nrow(dat6)
rows_removed = rows_all - rows_new
```

Upon filtering the original `articles` data frame, a total of 62 rows were removed.

6.A. Plot the amount of emotion words (the 8 from nrc) as a percentage of all the emotion words used each day (aggregate text from articles published on the same day).

```r
# load in stop words
data(stop_words)
```

```r
words <- dat6 %>%
  unnest_tokens(output = word, input = Text, token = 'words') %>%
  anti_join(stop_words)
```

```r
# remove all numbers
clean_words <- str_remove_all(words$word, "[:digit:]")

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

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

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

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

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

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

# remove possessive and replace with blank
clean_words <- gsub("'s", '', clean_words)
```

```r
# put the cleaned tokens into the `words` dataframe
# create `clean` column
words$clean <- clean_words

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

#reassign
words <- tib
```

```r
# nrc sentiment words
nrc <- get_sentiments('nrc') %>%
  filter(!sentiment %in% c("positive", "negative"))

# join nrc sentiment and words dataframes
nrc_sent <- words %>%
  inner_join(nrc) %>%
  na.omit() %>%
  group_by(Date, sentiment) %>%
  count() %>% # count number of sentiment words per day
  ungroup() %>%
  group_by(Date) %>%
  mutate(n_max_day = sum(n),
         percent = round((n/n_max_day)*100, 2))    # add total word
```

```r
plot1 <- nrc_sent %>%
  ggplot(aes(x = Date, y = percent, color = sentiment)) +
  geom_smooth(se = FALSE) +
  labs(x = "Date",
       y = "Frequency",
       title = "Proportion of sentiment of the term 'invasive species'",
       caption = "Data source: Nexus Uni") +
  theme_minimal()

plot1
```

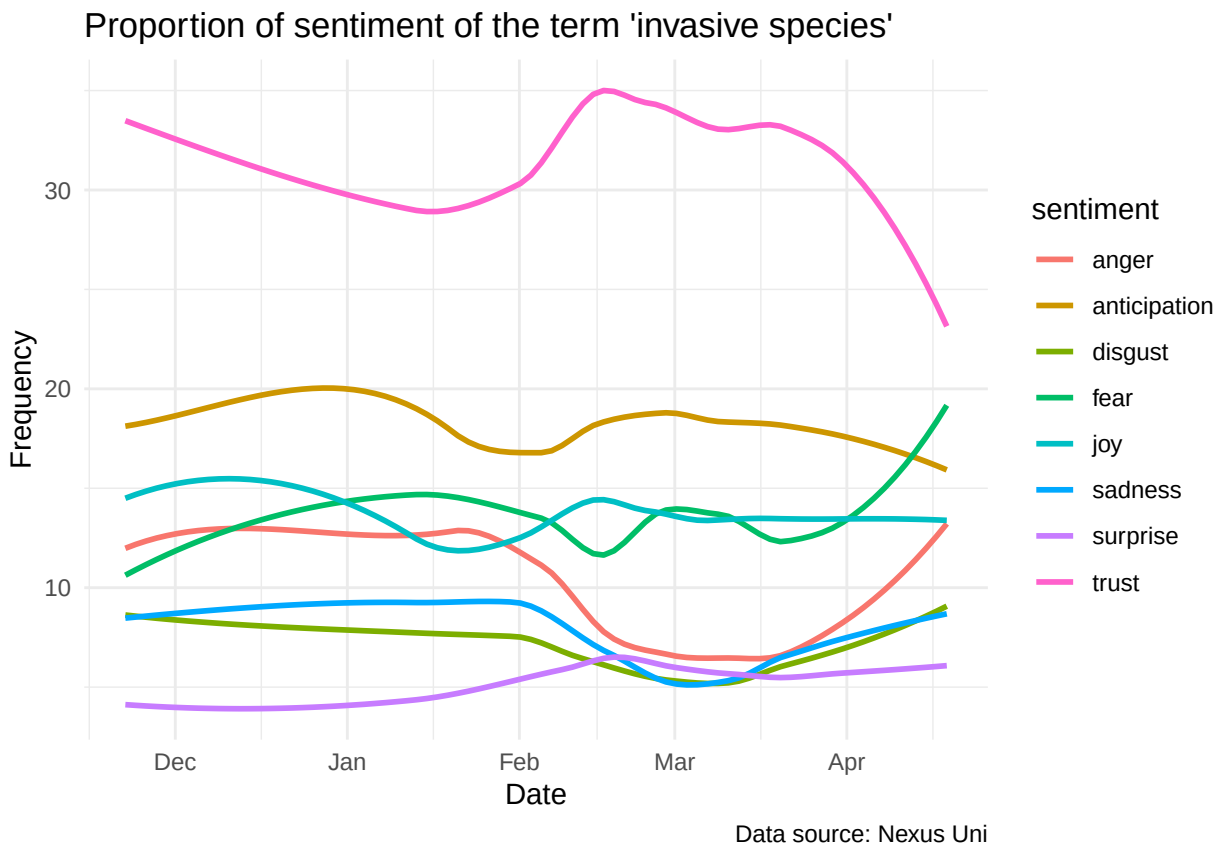

Figure 2: News media sentiment of the term 'invasive species' between April 17, 2021 & April 17, 2022

6.B. How does the distribution of emotion words change over time? Can you think of any reason this would be the case?

It seems as trust goes down, the sentiments of fear, anger, and distrust increase. There was possibly an outbreak of a new invasive species over this time.

References

Froehlich HE, Gentry RR, Rust MB, Grimm D, Halpern BS. (2017). Public Perceptions of Aquaculture: Evaluating Spatiotemporal Patterns of Sentiment around the World. PLoS ONE 12(1): e0169281. doi: 10.1371/journal.pone.0169281