EDS 231: Assignment 1

Scout Leonard

04/12/2022

Objective

In this first assignment for EDS231: text and sentiment analysis for environmental data science, I am applying skills learned in lab to pull data from the New York Times (NYT) database via thier API, then running some basic string manipulations and creating some basic plots based on a topic of interest from NYT articles. I have chosen to analyze NYT documetns which utilize the word "honeybee."

Load Libraries

Assignment 1 Responses

Part 1

Create a free New York Times account (https://developer.nytimes.com/get-started)

Done! I have my key saved in a safe place (in addition to the place where it appears in my code below, haha!)

Part 2

Pick an interesting environmental key word(s) and use the jsonlite package to query the API. Pick something high profile enough and over a large enough time frame that your query yields enough articles for an interesting examination.

I chose the keyword, "honeybee," and used data starting in 2017 (3 years of data) until today.

In the code chunk below, I set my search parameters and call data from the NYT API using my key, so that I can pull all the publications from these dates with my environmental key word.

The code will return the URL by search parameters build.

```
#set search parameters
term <- "honeybee" # Need to use + to string together separate words
begin_date <- "20190101"
end_date <- "20220410"
key <- "CW2nwn63au0uAelc9XjRMGfQ1XZ2Vpo9"

#base url using our start and end dates and key term
baseurl <- paste0("http://api.nytimes.com/svc/search/v2/articlesearch.json?q=",term,
                  "&begin_date=",begin_date,"&end_date=",end_date,
                  "&facet_filter=true&api-key=", key, sep="")

#check: return base url
baseurl
```

```
## [1] "http://api.nytimes.com/svc/search/v2/articlesearch.json?q=honeybee&begin_date=20190101&end_date=
```

Part 3

Recreate the publications per day and word frequency plots using the first paragraph

```
## [1] "data.frame"
```

Publications Per Day Plot

```
nytDat %>%
  mutate(pubDay = gsub("T.*",
                       "",
                       response.docs.pub_date)) %>%
  group_by(pubDay) %>%
  summarise(count=n()) %>%
  filter(count >= 2) %>%
  ggplot() +
  geom_bar(aes(x=reorder(pubDay, count),
               y=count),
           stat="identity") +
  coord_flip() +
  labs(title = "NYT Publications Per Day with 'honeybee' in the publication")
```

Word Frequency Plot

```
#check that this returns the lead paragraph,ie returns response.doc.lead.paragraph
paragraph <- names(nytDat)[6] #The 6th column, "response.doc.lead_paragraph", is the one we want here.

tokenized <- nytDat %>%
  unnest_tokens(word, paragraph)
```

Below, I create a word frequency plot without stemmed key terms or removing numbers, just removing stop words. This is for comparison so that I know that my processing of the tokenized data outputs changes as expected, altering the distribution of word frequencies.

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

#remove stop words from tokenized first paragraph words
tokenized <- tokenized %>%
  anti_join(stop_words)
```

```
#count occurrences of all the words in tokenized, and plot all the words that occur more than 10 times
tokenized %>%
  count(word, sort = TRUE) %>%
  filter(n > 5) %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(n, word)) +
  geom_col() +
  labs(y = NULL,
       title = "Frequency of words co-occuring with 'honeybee' in NYT publications 1st paragraph")
```

Make some (at least 3) transformations to the corpus (add stopword(s), stem a key term and its variants, remove numbers)

```
#remove numbers
clean_tokens <- str_remove_all(tokenized$word, "[:digit:]")
```

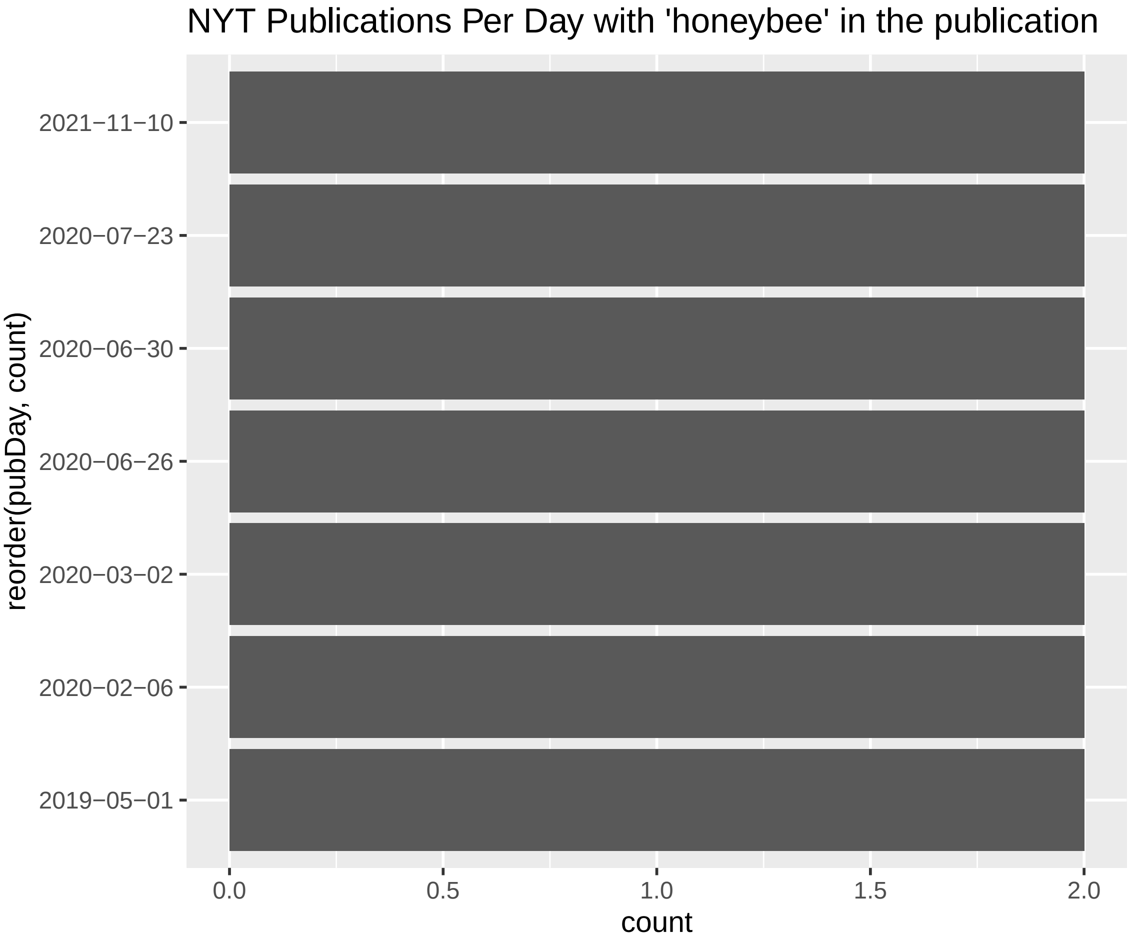


Figure 1: The figure shows the frequency of the use of my environmental key word, ‘honeybee,’ with the dates on the y axis showing the date for which the corresponding x axis value, a frequncy of mentions per day, corresponds to.

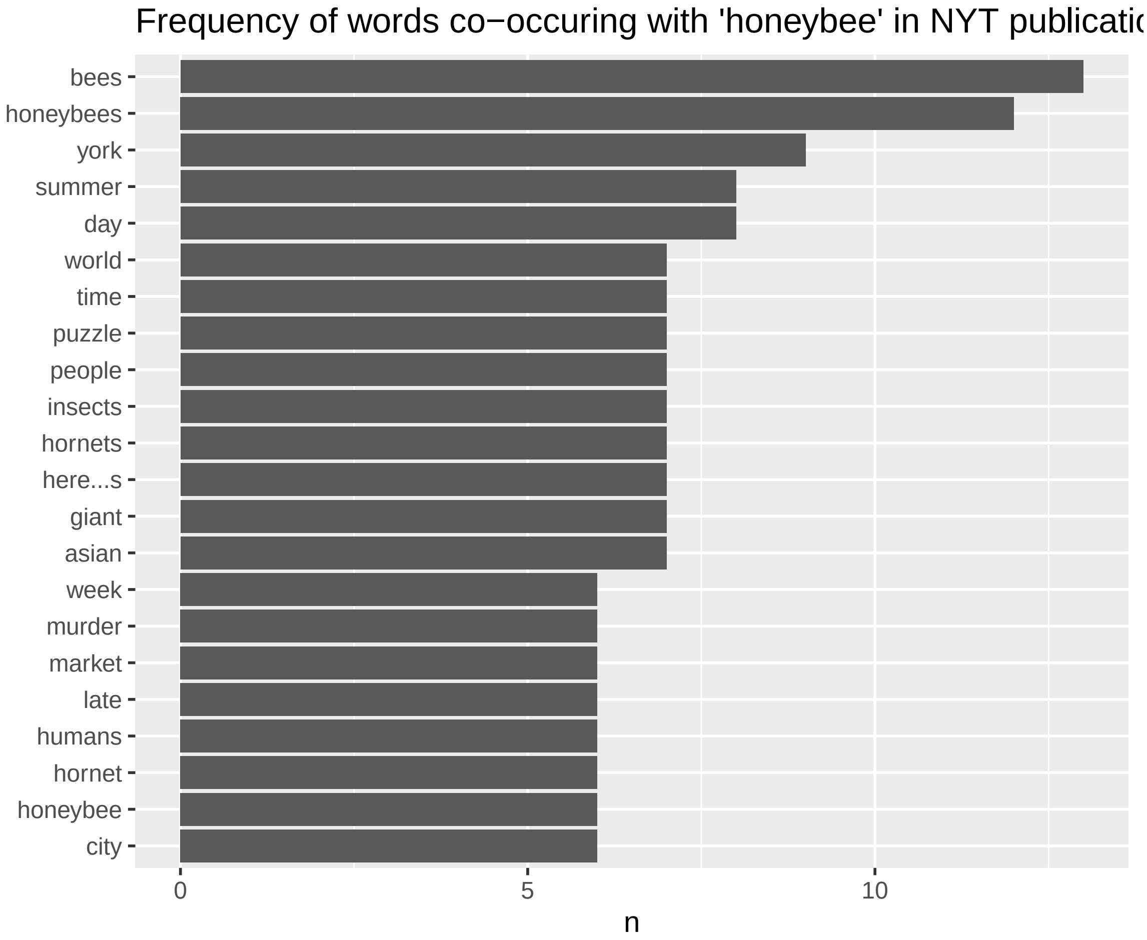


Figure 2: A frequency word plot of the use of ‘honeybee’ in New York Times publications first paragraph. No processing of tokenized data has yet occured in for the word data, except for removing stop words with the R dataset ‘stop_words’.

```
#removes appostrophes
clean_tokens <- gsub("’s", '', clean_tokens)

#add stopwords
clean_tokens <- str_remove_all(clean_tokens, "ago")
clean_tokens <- str_remove_all(clean_tokens, "nytimes.com") # i notice a lot of words from the new york
clean_tokens <- str_remove_all(clean_tokens, "york")
clean_tokens <- str_remove_all(clean_tokens, "it's")
clean_tokens <- str_remove_all(clean_tokens, "york")
clean_tokens <- str_remove_all(clean_tokens, "times")
clean_tokens <- str_remove_all(clean_tokens, "week") # also notive a lot of time words which i think ar
clean_tokens <- str_remove_all(clean_tokens, "day")
clean_tokens <- str_remove_all(clean_tokens, "time")

#stem a key term - bee words
clean_tokens <- str_replace_all(clean_tokens,"honeybee[a-z,A-Z]*","honeybee") #stem honeybee to honeybe
clean_tokens <- str_replace_all(clean_tokens,"bee[a-z,A-Z]*","bee")
clean_tokens <- str_replace_all(clean_tokens,"hive[a-z,A-Z]*","hive")
clean_tokens <- str_replace_all(clean_tokens,"hornet[a-z,A-Z]*","hornet")

tokenized$clean <- clean_tokens

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

#reassign
tokenized <- tib

#try again
tokenized %>%
  count(clean, sort = TRUE) %>%
  filter(n > 4) %>% #illegible with all the words displayed
  mutate(clean = reorder(clean, n)) %>%
  ggplot(aes(n, clean)) +
  geom_col() +
  labs(y = NULL)
```

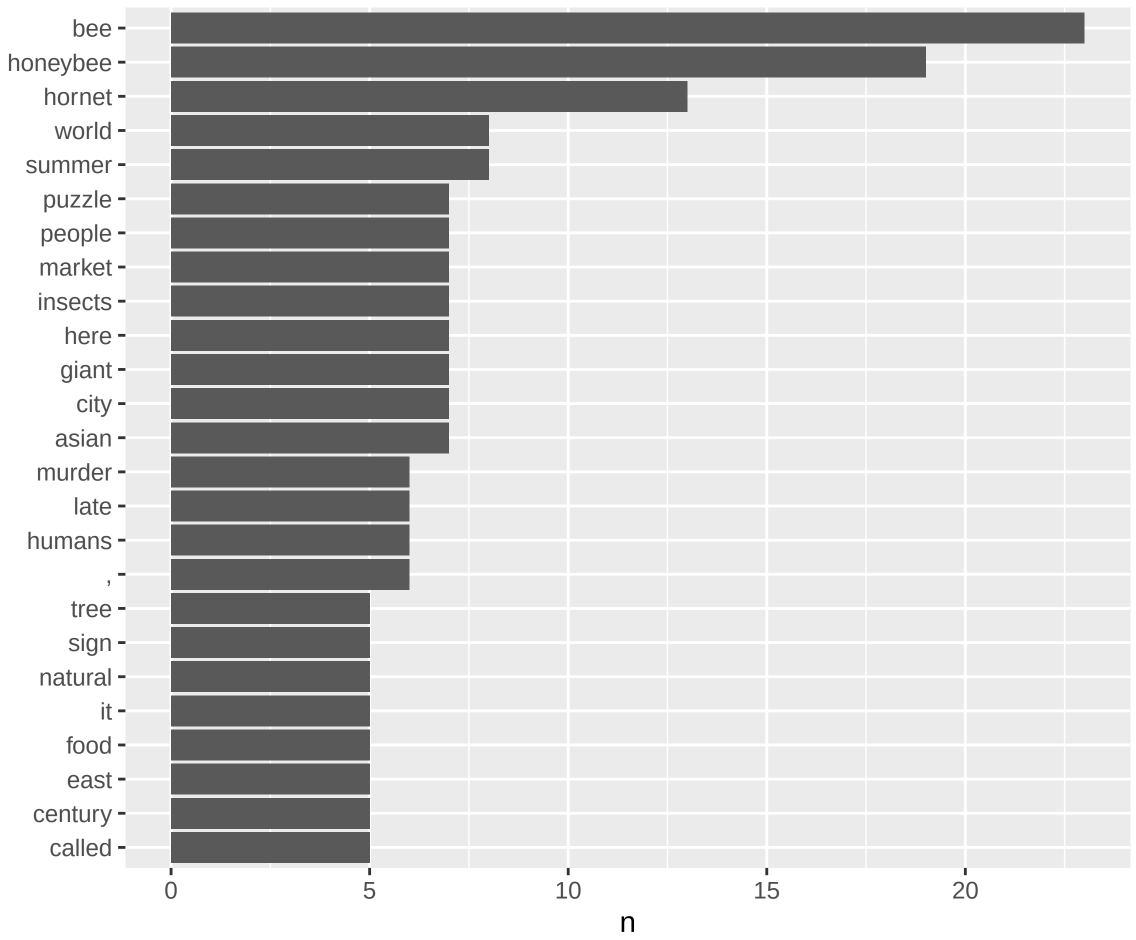


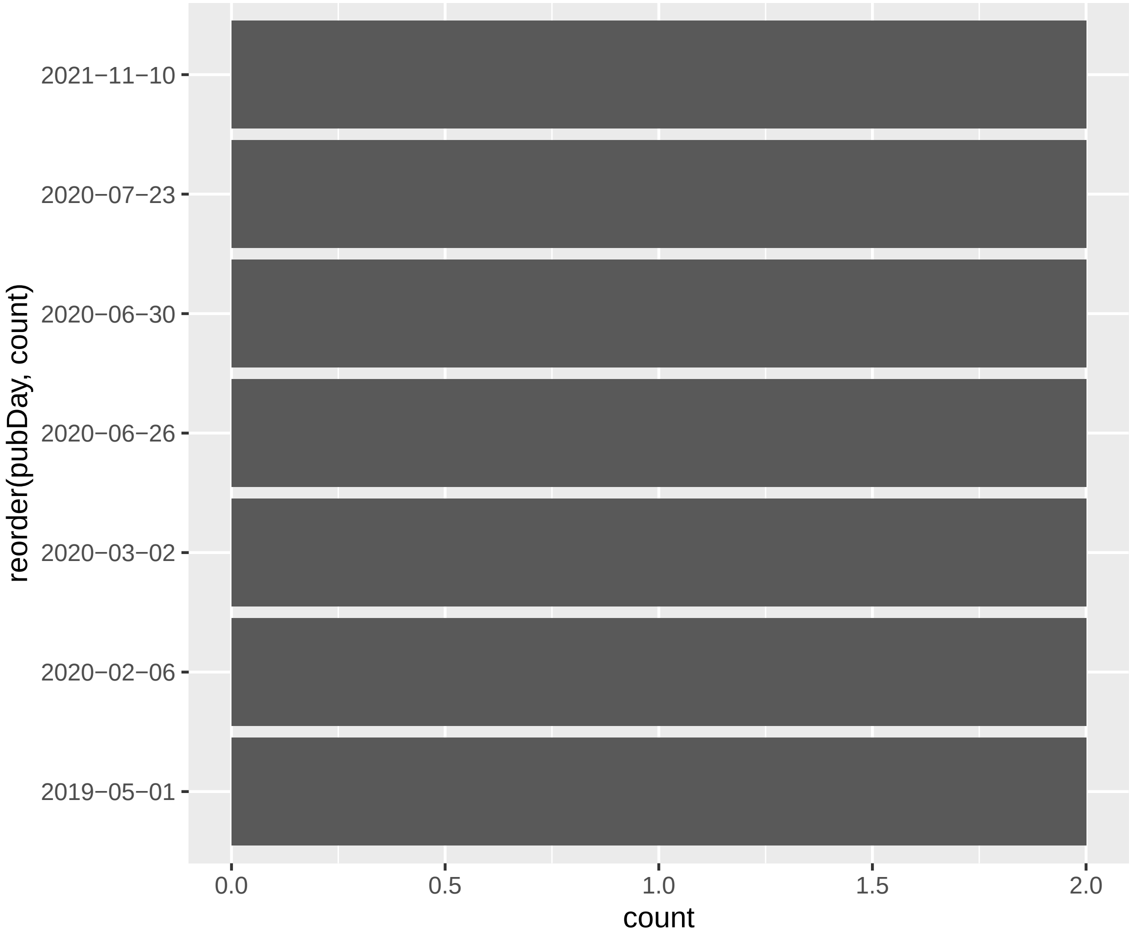
Figure 3: A frequency word plot of the use of 'honeybee' in New York Times publications first paragraphs. Now processing of tokenized data has occured for the word data, including removing stop words with the R dataset 'stop_words', stemming key terms, and removing numbers.

Part 4

Recreate the publications per day and word frequency plots using the headlines variable (response.docs.headline.main). Compare the distributions of word frequencies between the first paragraph and headlines. Do you see any difference?

Publications Per Day Plot

```
nytDat %>%
  mutate(pubDay=gsub("T.*","",response.docs.pub_date)) %>%
  group_by(pubDay) %>%
  summarise(count=n()) %>%
  filter(count >= 2) %>%
  ggplot() +
  geom_bar(aes(x=reorder(pubDay, count), y=count), stat="identity") + coord_flip()
```



Word frequency plot

```
#check that this returns the lead paragraph,ie returns response.doc.lead.paragraph
headline <- names(nytDat)[21] #The 6th column, "response.doc.lead_main", is the one we want here.

headlines_tokenized <- nytDat %>%
  unnest_tokens(word, headline)
```

```r
data(stop_words)

headlines_tokenized <- headlines_tokenized %>%
  anti_join(stop_words) #removes the stop words
```

```r
#make frequency plot
headlines_tokenized %>%
  count(word, sort = TRUE) %>%
  filter(n > 5) %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(n, word)) +
  geom_col() +
  labs(y = NULL,
       title = "Frequency of words co-occuring with 'honeybee' in NYT headlines")
```

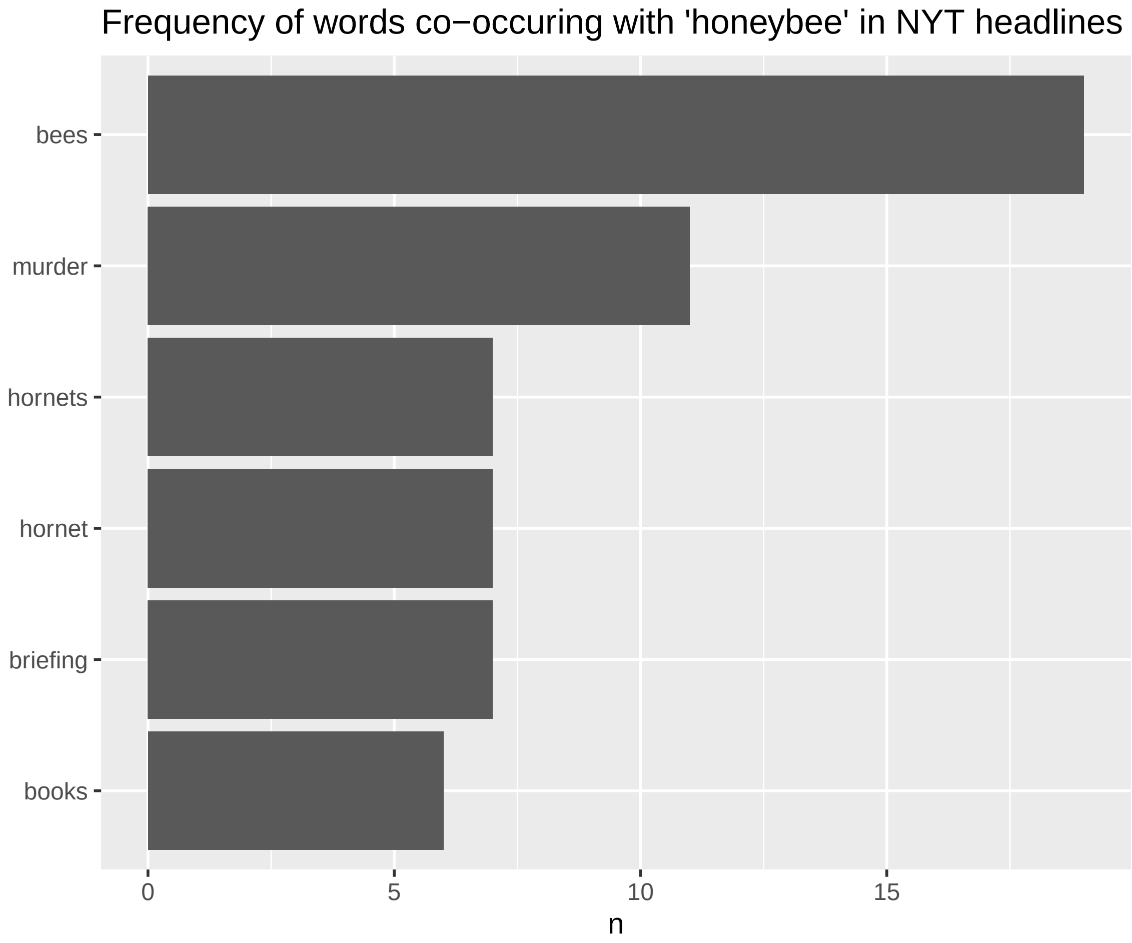


Figure 4: A frequency word plot of the use of ‘honeybee’ in New York Times publications headlines. No processing of tokenized data has yet occured in for the word data, except for removing stop words with the R dataset ‘stop_words’.

```r
#remove numbers
clean_headline_tokens <- str_remove_all(headlines_tokenized$word, "[:digit:]")

#removes apostrophes
clean_tokens <- gsub("’s", '', clean_headline_tokens)
```

```
#add stopwords
clean_headline_tokens <- str_remove_all(clean_headline_tokens, "ago")
clean_headline_tokens <- str_remove_all(clean_headline_tokens, "nytimes.com") # i notice a lot of words
clean_headline_tokens <- str_remove_all(clean_headline_tokens, "york")
clean_headline_tokens <- str_remove_all(clean_headline_tokens, "it's")
clean_headline_tokens <- str_remove_all(clean_headline_tokens, "york")
clean_headline_tokens <- str_remove_all(clean_headline_tokens, "times")
clean_headline_tokens <- str_remove_all(clean_headline_tokens, "week") # also notice a lot of time word
clean_headline_tokens <- str_remove_all(clean_headline_tokens, "day")
clean_headline_tokens <- str_remove_all(clean_headline_tokens, "time")
clean_headline_tokens <- str_remove_all(clean_headline_tokens, "briefing")
clean_headline_tokens <- str_remove_all(clean_headline_tokens, "books")

#stem a key term - bee words
clean_headline_tokens <- str_replace_all(clean_headline_tokens,"honeybee[a-z,A-Z]*","honeybee") #stem h
clean_headline_tokens <- str_replace_all(clean_headline_tokens,"bee[a-z,A-Z]*","bee")
clean_headline_tokens <- str_replace_all(clean_headline_tokens,"hive[a-z,A-Z]*","hive")
clean_headline_tokens <- str_replace_all(clean_headline_tokens,"hornet[a-z,A-Z]*","hornet")

headlines_tokenized$clean <- clean_headline_tokens

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

#reassign
headlines_tokenized <- tib

#try again with processsed word data
headlines_tokenized %>%
  count(clean, sort = TRUE) %>%
  filter(n > 3) %>% #illegible with all the words displayed
  mutate(clean = reorder(clean, n)) %>%
  ggplot(aes(n, clean)) +
  geom_col() +
  labs(y = NULL)
```

I definitely see a difference. Something troubling that I notice is that for the frequency of words co-occuring with honeybee in the first paragraph of publications, the frequency of the words “asian,” “murder,” and “hornet” are more frequent than other words I might associate with bees (pollination, food - which does make the cut, agriculture, colony collapse, etc.) or even less racist alarmist terms for the invasive insect *Vespa mandarinia*, like invasive, damage, local, etc. It seems these words were frequently used when covering honeybees in the last three years, which is extra damaging in the time of rising anti-American hate crimes during the COVID-19 pandemic. I was surprised that coverage of the invasive *Vespa mandarinia* has this impact on coverage of honeybees, considering the many environmental threats honeybees continue to face.

“Murder” and “hornet” still found their way onto the word frequency plot for headlines, but they are joined by “swarm,” an expected co-occuring word, and “u.s.” and “evening.”

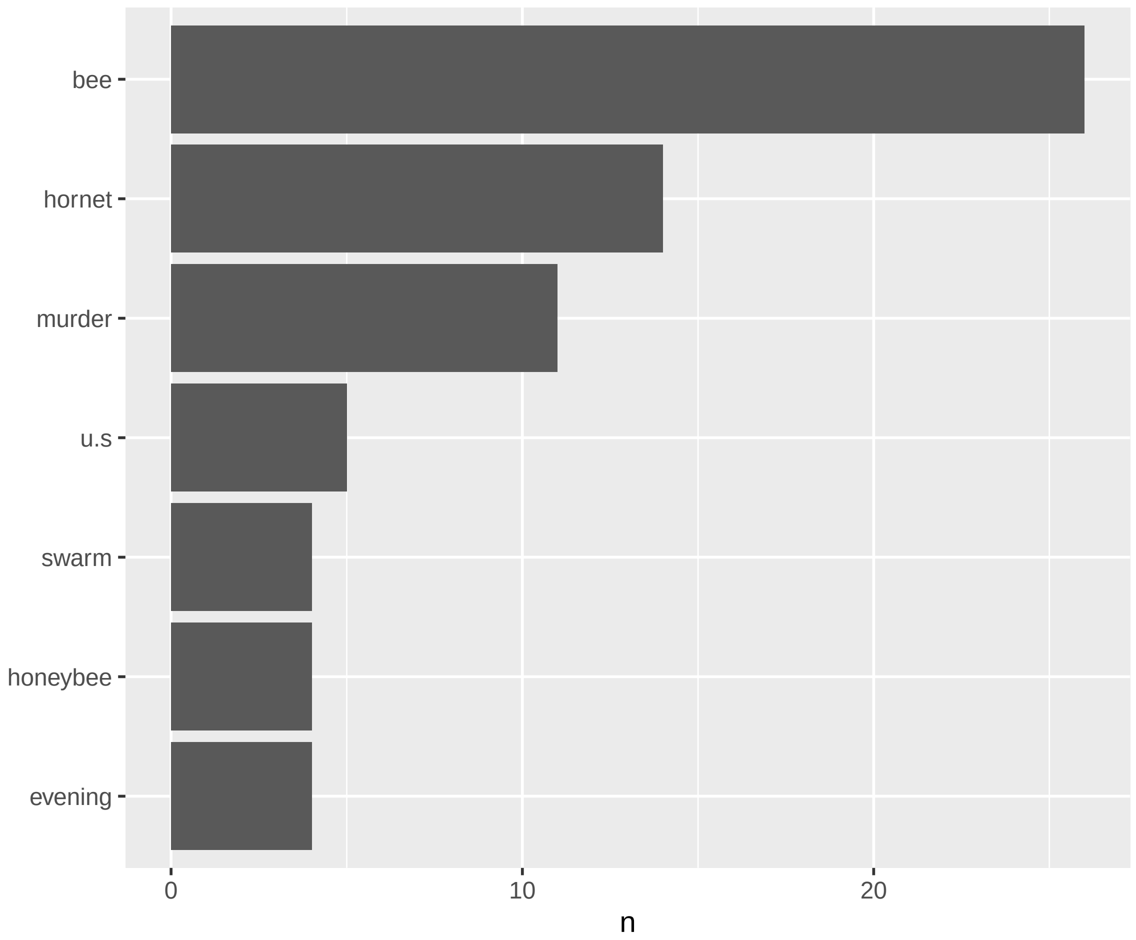


Figure 5: A frequency word plot of the use of 'honeybee' in New York Times publications headlines. Now processing of tokenized data has occured for the word data, including removing stop words with the R dataset 'stop_words', stemming key terms, and removing numbers.