Tweet Storm on the Horizon

Overview

President Trump, love him or hate him or just don't care, is the most famous & powerful Twitter user of all time. Like it or not, President Trump's tweets have become a source of information. The New York Times, Wall Street Journal and others news outlets and take a look at President Trump's Tweets why shouldn't we? This week let's put politics aside and let the data do the talking! You can either use my dataset "TrumpQ12020Tweets.csv" or grab the latest tweets from the http://www.trumptwitterarchive.com/. Just be sure to grab enough to do some analysis - i.e. 3 months or more.

Load Libraries

library(tidyverse) library(lubridate) library(tidytext) library(topicmodels) library(wordcloud2)

```
options(warn = -1)
library(tidyverse)
library(lubridate)
library(tidytext)
library(topicmodels)
library(wordcloud2)
```

load tweets

Note the data is pipe delimited(delim = "|") so you'll need to read them with read_delim instead of read_csv, if you read ahead you'll also see that you might need to transform created_at as a date variable (col_types = cols(created_at = col_datetime(format = "%m-%d-%Y %H:%M:%S")))

"TrumpQ12020Tweets.csv"

```
tweet <- read_delim("TrumpQ12020Tweets.csv") %>%
  mutate(created_at = as.Date(created_at, format = "%m-%d-%Y %H:%M:%S"))

tweet
```

```
## # A tibble: 2,783 x 7
##    source             text             created_at retwe~1 favor~2 is_re~3 id_str
##    <chr>              <chr>            <date>     <chr>     <dbl> <chr>   <chr>
##  1 Twitter for iPhone Will be intervi~ 2020-03-30 15419     94155 false   12444~
##  2 Twitter for iPhone RT @SteveFDA: W~ 2020-03-30 6913          0 false   12444~
##  3 Twitter for iPhone RT @GovMikeDeWi~ 2020-03-30 9050          0 false   12444~
##  4 Twitter for iPhone https://t.co/Yz~ 2020-03-29 13735     65496 false   12444~
##  5 Twitter for iPhone https://t.co/Mt~ 2020-03-29 14134     60385 false   12444~
##  6 Twitter for iPhone RT @WhiteHouse:~ 2020-03-29 10583         0 false   12443~
##  7 Twitter for iPhone Will be startin~ 2020-03-29 13123     85698 false   12443~
```

```
##  8 Twitter for iPhone So proud of the~ 2020-03-29 16015     70179 false   12443~
##  9 Twitter for iPhone Thank you very ~ 2020-03-29 9513      54266 false   12443~
## 10 Twitter for iPhone I am a great fr~ 2020-03-29 99226    537499 false   12443~
## # ... with 2,773 more rows, and abbreviated variable names 1: retweet_count,
## #   2: favorite_count, 3: is_retweet
```

Term Frequency & Wordcloud

create tweet_freq table

1. create a month_variable
2. parse terms into words, remove the following

- stop words
- c("t.co", "https", "false", "twitter", "iphone", "amp", "rt", "android")

3. summarize by month and word
4. take top 100 words by month

create the following word clouds: 1. word cloud for month 1 2. word cloud for month 2 3. word cloud for month 3

answer: what terms jump out at you?

```
tweet_freq <-
  tweet %>%
  mutate(month = month(created_at, label=TRUE, abbr=FALSE)) %>%
  filter(!is.na(month)) %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words, by = c("word" = "word")) %>%
  filter(!word %in% c("t.co", "https", "false", "twitter", "iphone", "amp", "rt", "android")) %>%
  filter(!str_detect(word, "^\\d")) %>%
  group_by(month, word) %>%
  summarize(n = n()) %>%
  ungroup() %>%
  group_by(month) %>%
  slice_max(order_by = n, n = 100)

tweet_freq
```

```
## # A tibble: 323 x 3
## # Groups:   month [3]
##    month   word               n
##    <ord>   <chr>          <int>
##  1 January democrats        135
##  2 January realdonaldtrump  128
##  3 January impeachment      125
##  4 January president        124
##  5 January house             84
##  6 January senate            70
##  7 January schiff            66
##  8 January people            61
```

```
##  9 January american           60
## 10 January trump              60
## # ... with 313 more rows
```

```
wordcloud <- function(m){
  tweet_freq %>%
  ungroup() %>%
  filter(month == m) %>%
  select(word, n) %>%
  wordcloud2(size = 0.5)
}

#wordcloud("January")
#wordcloud("February")
#wordcloud("March")

for (c in c("January", "February", "March")){
  print(wordcloud(c))
}

# answer: what terms jump out at you?
# Month 1: democrats, realdonaldtrump, impeachment, ......
# Month 2: realdonaldtrump, president, trump, ......
# Month 3: realdonaldtrump, coronavirus, people, ......
```

Bigram Analysis

create table bigram_freq by 1. create a bigram
2. use separate to split bigram into word1 and word2 then filter the following - stop words against both word1 and word2 - c("t.co", "https", "false", "twitter", "iphone", "amp", "rt", "android") - filter digits 3. create a bigram variable by combining word1 and word2 together 4. count the bigram up.

create the following

1. make a chart of the top 10 terms that come after the word "fake", like: "fake news"
2. make a chart of the top 10 terms that come before the word "media", like: "mainstream media"
3. make a chart of the top 5 terms that contain the word "joe", like "joe biden" or "sleepy joe"

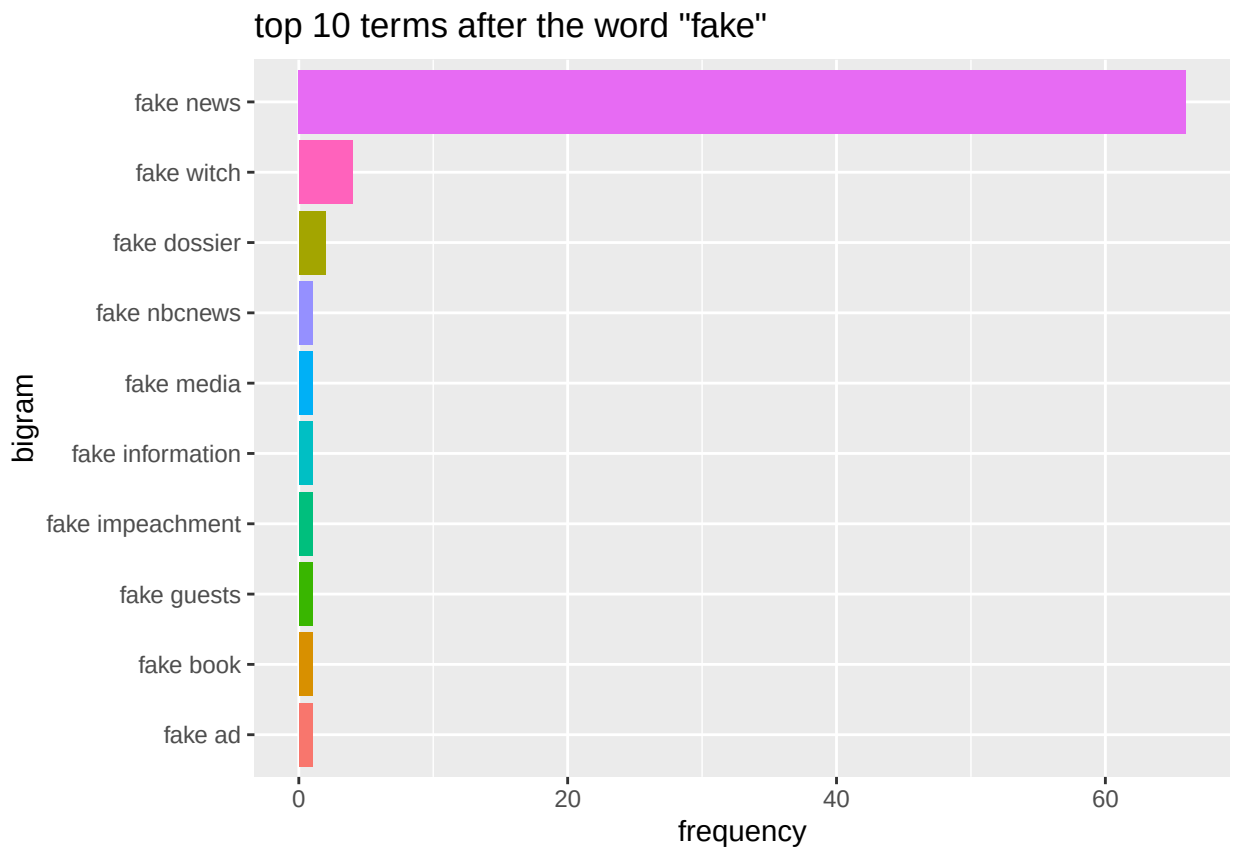
answer: what jumps out at you?

```
bigram_freq <-
  tweet %>%
  unnest_tokens(bigram, text, token = "ngrams", n = 2, n_min = 2) %>%
  separate(bigram, c("word1", "word2"), sep = " ") %>%
  filter(!word1 %in% stop_words$word) %>%
  filter(!word2 %in% stop_words$word) %>%
  filter(!word1 %in% c("t.co", "https", "false", "twitter", "iphone", "amp", "rt", "android")) %>%
  filter(!word2 %in% c("t.co", "https", "false", "twitter", "iphone", "amp", "rt", "android")) %>%
  filter(!str_detect(word1, "^\\d")) %>%
  filter(!str_detect(word2, "^\\d")) %>%
  unite(bigram, word1, word2, sep = " ") %>%
```

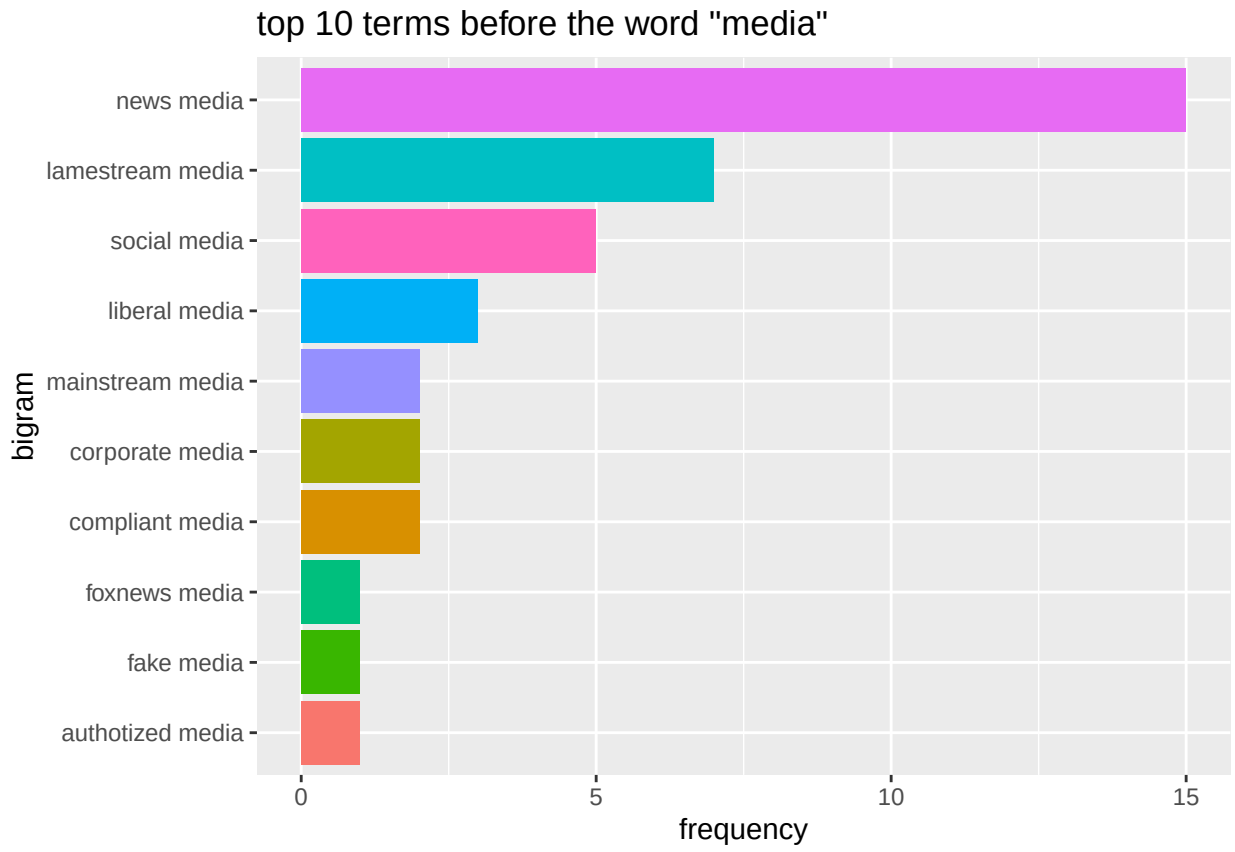
```
  count(bigram, sort = TRUE)

bigram_freq
```

```
## # A tibble: 8,078 x 2
##    bigram                       n
##    <chr>                    <int>
##  1 president realdonaldtrump    89
##  2 fake news                    66
##  3 mini mike                    58
##  4 president trump              53
##  5 impeachment hoax             47
##  6 american people              44
##  7 white house                  43
##  8 republican party             35
##  9 adam schiff                  34
## 10 joe biden                    28
## # ... with 8,068 more rows
```

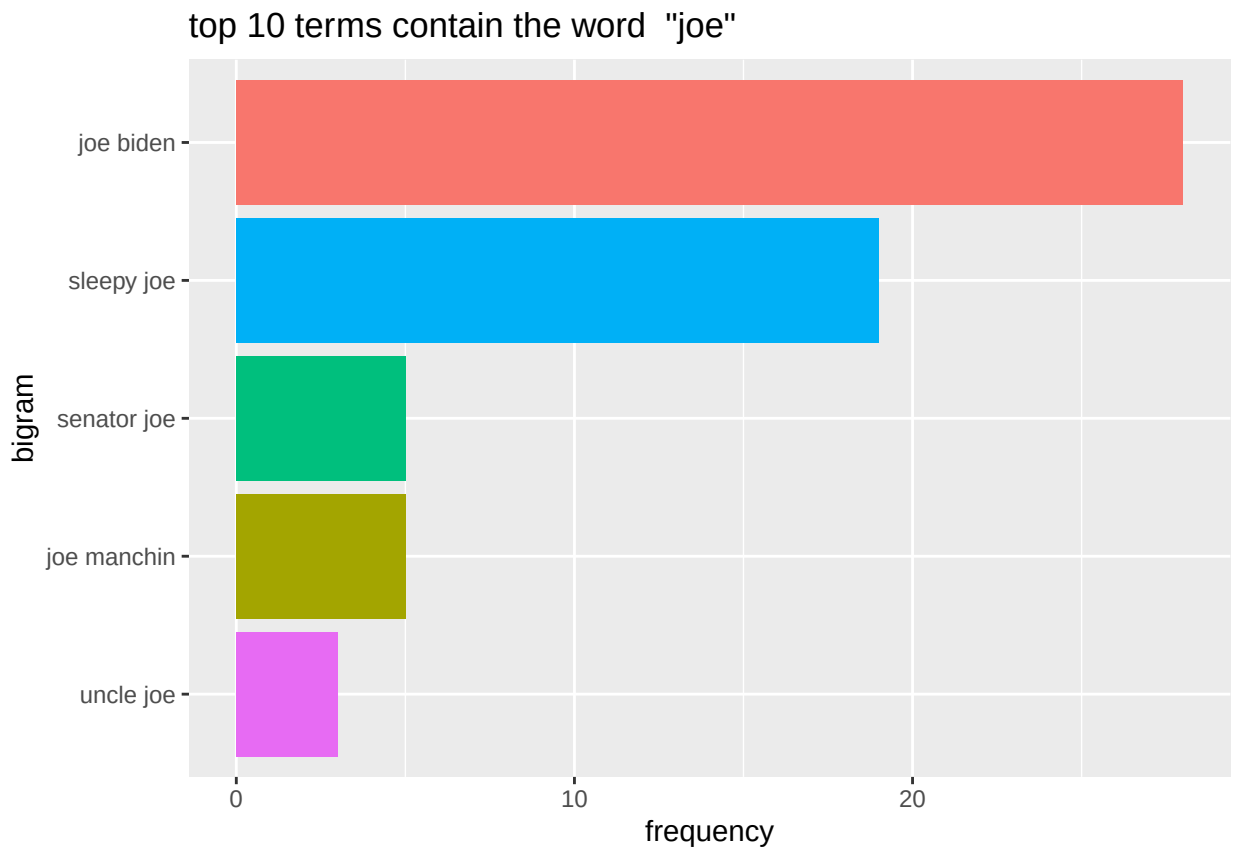
```
bigram_freq %>%
  separate(bigram, c("word1", "word2"), sep = " ") %>%
  filter(word1 == "fake") %>%
  unite(bigram, word1, word2, sep = " ") %>%
  slice_max(order_by = n, n = 10) %>%
  head(10) %>%
  ggplot() +
  geom_col(mapping = aes(x=n, y=reorder(bigram,n), fill=factor(bigram))) +
  labs(title = 'top 10 terms after the word "fake"', x = 'frequency', y = 'bigram') +
  theme(legend.position = "none")
```




```
bigram_freq %>%
  separate(bigram, c("word1", "word2"), sep = " ") %>%
  filter(word2 == "media") %>%
  unite(bigram, word1, word2, sep = " ") %>%
  slice_max(order_by = n, n = 10) %>%
  head(10) %>%
  ggplot() +
  geom_col(mapping = aes(x=n, y=reorder(bigram,n), fill=factor(bigram))) +
  labs(title = 'top 10 terms before the word "media"', x = 'frequency', y = 'bigram') +
  theme(legend.position = "none")
```




```
bigram_freq %>%
  separate(bigram, c("word1", "word2"), sep = " ") %>%
  filter(word1 == "joe" | word2 == "joe") %>%
  unite(bigram, word1, word2, sep = " ") %>%
  slice_max(order_by = n, n = 5) %>%
  head(5) %>%
  ggplot() +
  geom_col(mapping = aes(x=n, y=reorder(bigram,n), fill=factor(bigram))) +
  labs(title = 'top 10 terms contain the word  "joe"', x = 'frequency', y = 'bigram') +
  theme(legend.position = "none")
```




```
# answer: what jumps out at you?
# As for the terms after the word "fake", Trump says "fake news" the most. As for the terms before the
```

Sentiments

create sentiment_by_month 1. inner join words_by_month to "bing" sentiments 2. group by month and sentiment 3. get the top 10 words by sentiment by month ~ group by (sentiment, month) then slice_max() 4. make words with negative sentiment negative (-n) and positive words positive

create the following bar charts

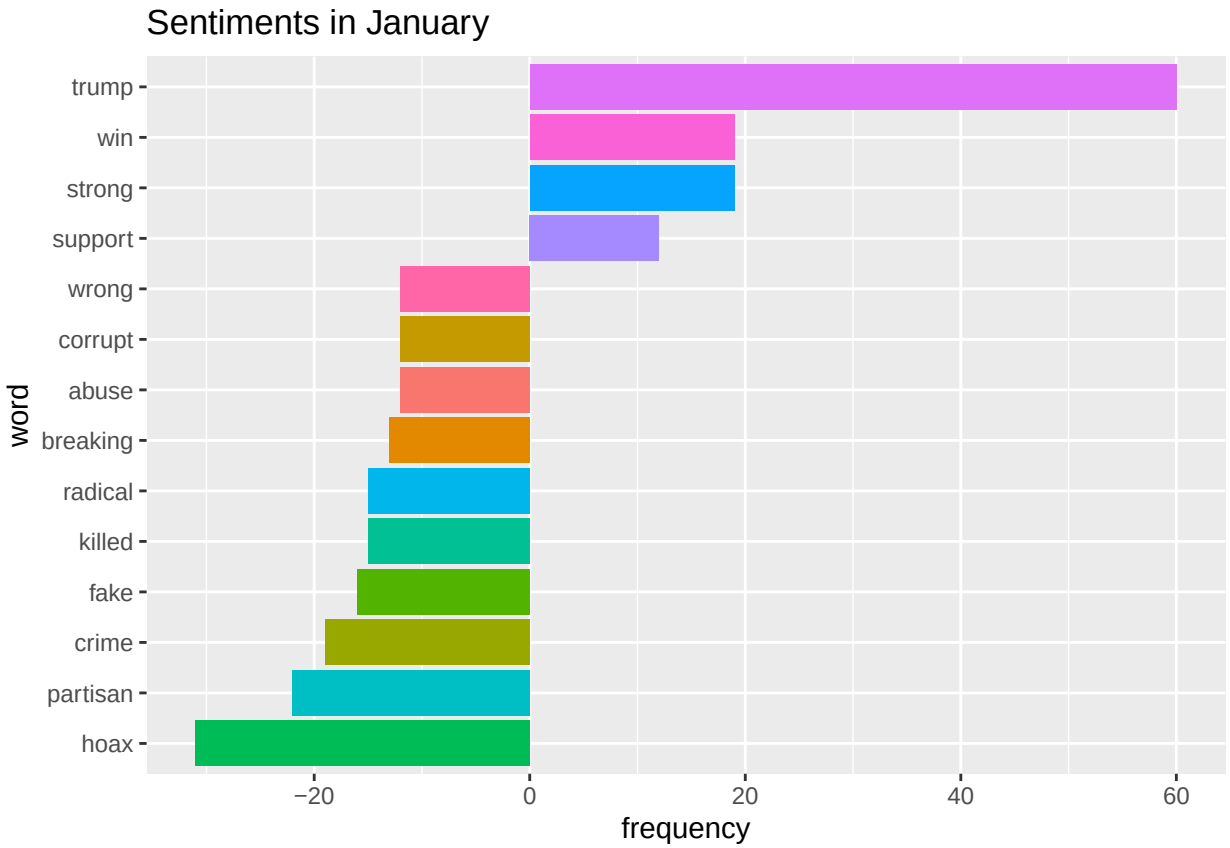
1. chart 1 sentiment for month 1, be sure to order n, and coord_flip
2. chart 1 sentiment for month 2, be sure to order n, and coord_flip
3. chart 1 sentiment for month 3, be sure to order n, and coord_flip

Answer: what if anything does this tell you? did the sentiment change month to month?

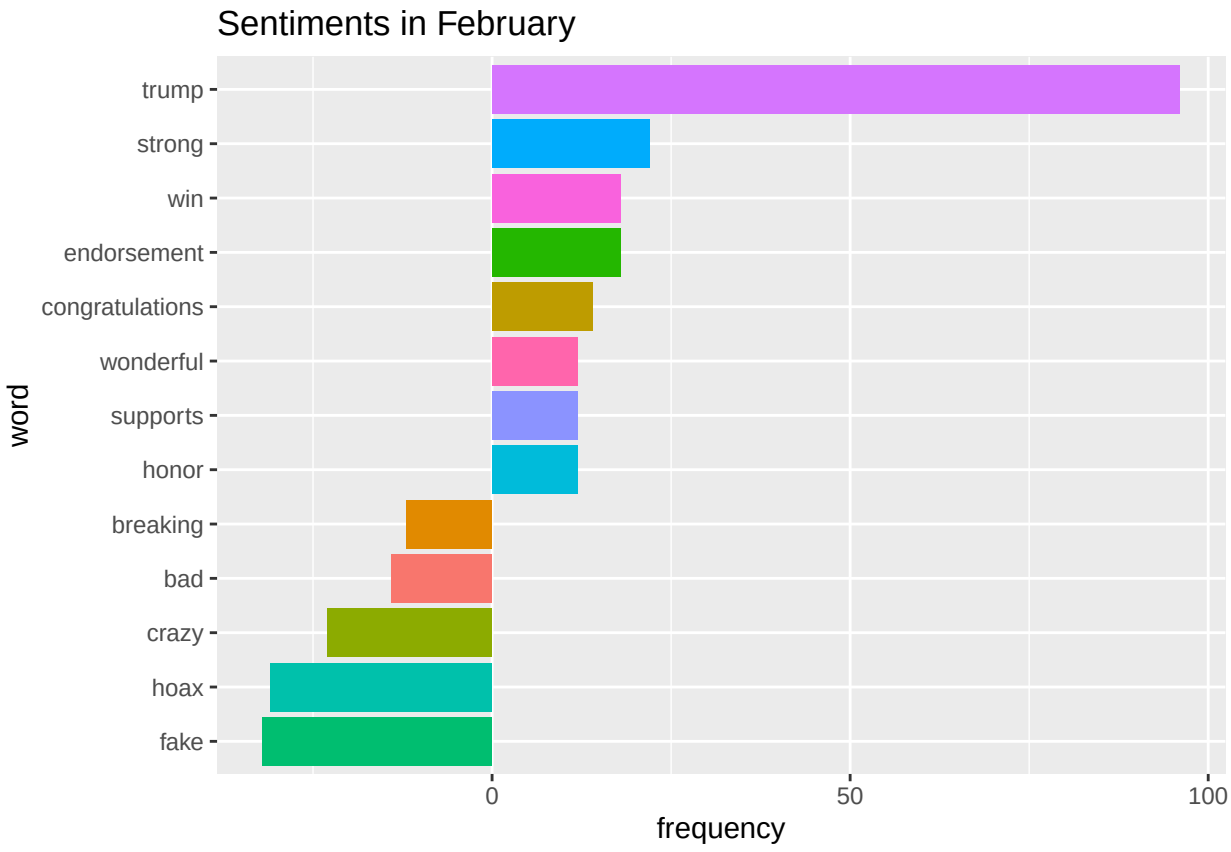
```
sentiment_by_month <-
  tweet_freq %>%
  inner_join(get_sentiments("bing")) %>%
  group_by(month, sentiment) %>%
  slice_max(order_by = n, n = 10) %>%
  mutate(n=if_else(sentiment=='negative',-n,n))

sentiment_by_month
```

```
## # A tibble: 41 x 4
## # Groups:   month, sentiment [6]
##    month   word          n sentiment
##    <ord>   <chr>     <int> <chr>
##  1 January hoax        -31 negative
##  2 January partisan    -22 negative
##  3 January crime       -19 negative
##  4 January fake        -16 negative
##  5 January killed      -15 negative
##  6 January radical     -15 negative
##  7 January breaking    -13 negative
##  8 January abuse       -12 negative
##  9 January corrupt     -12 negative
## 10 January wrong       -12 negative
## # ... with 31 more rows
```

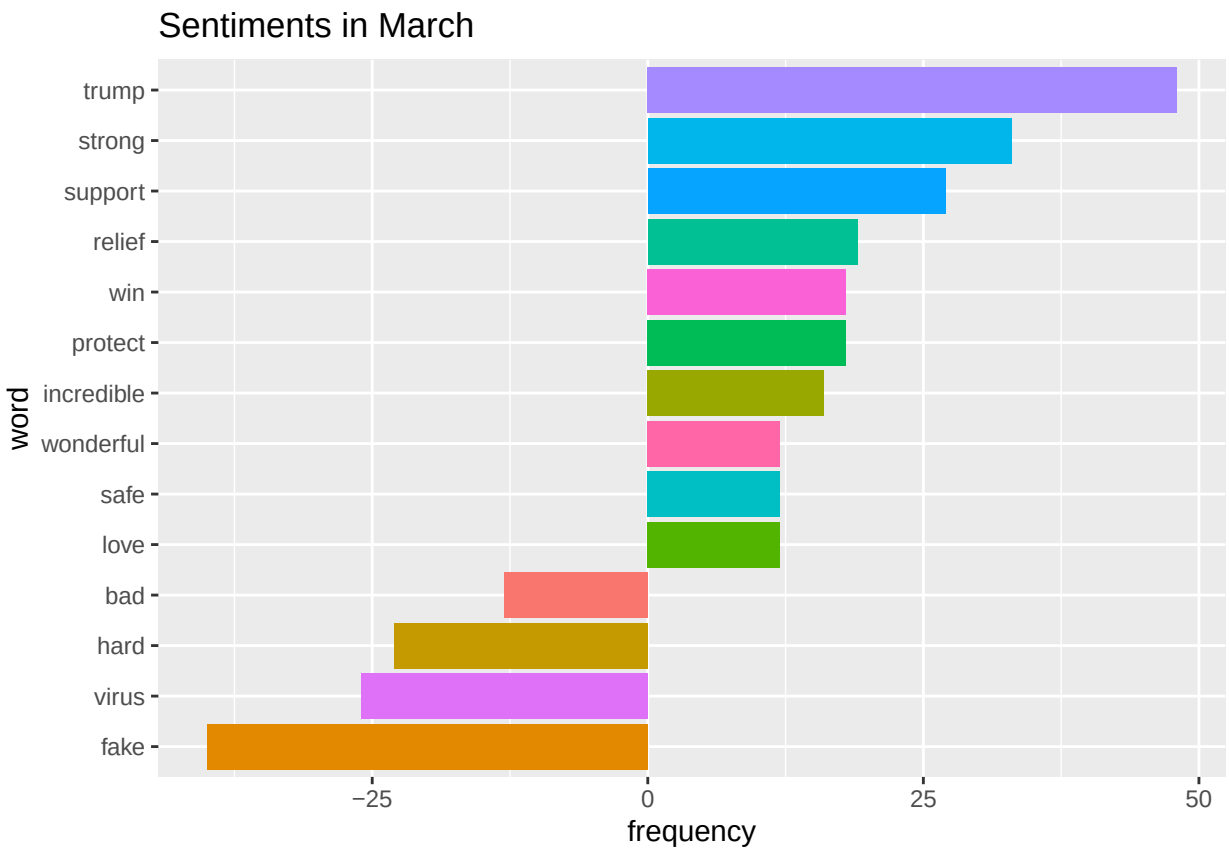
```
sentiment_by_month %>%
  filter(month == "January") %>%
  ggplot() +
  geom_col(mapping = aes(x=reorder(word,n), y=n, fill=factor(word))) +
  coord_flip() +
  labs(title = 'Sentiments in January', x = 'word', y = 'frequency') +
  theme(legend.position = "none")
```



```
sentiment_by_month %>%
  filter(month == "February") %>%
  ggplot() +
  geom_col(mapping = aes(x=reorder(word,n), y=n, fill=factor(word))) +
  coord_flip() +
  labs(title = 'Sentiments in February', x = 'word', y = 'frequency') +
  theme(legend.position = "none")
```




```
sentiment_by_month %>%
  filter(month == "March") %>%
  ggplot() +
  geom_col(mapping = aes(x=reorder(word,n), y=n, fill=factor(word))) +
  coord_flip() +
  labs(title = 'Sentiments in March', x = 'word', y = 'frequency') +
  theme(legend.position = "none")
```




```
# Answer: what if anything does this tell you? did the sentiment change month to month?
# Trump mentioned the words "trump", "strong", and "win", the most times in all three months. The senti
```

Topic Prep

Create tweet_dtm by preparing a Document Term Matrix (dtm)

1. unnest tokens into words
2. remove the following

- stop words
- c(“t.co”, “https”, “false”, “twitter”, “iphone”, “amp”, “rt”, “android”)

3. create a document id using tweet_id or similar unique identifier
4. count document, word order by count
5. cast the result to a document term matrix
6. use lda with the matrix to generate a model

create tweet_lda by taking your tweet_dtm, pick a value of k (4,6,8 or 10)

```
tweet_dtm <-
  tweet %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words) %>%
  filter(!word %in% c("t.co", "https", "false", "twitter", "iphone", "amp", "rt", "android")) %>%
  filter(!str_detect(word,"^\\d")) %>%
  mutate(document_id = id_str) %>%
  group_by(document_id,word) %>%
  count(word, sort = TRUE) %>%
  cast_dtm(document_id, word, n)

tweet_lda <- LDA(tweet_dtm, k = 4, control = list(seed = 1234))
tweet_lda
```

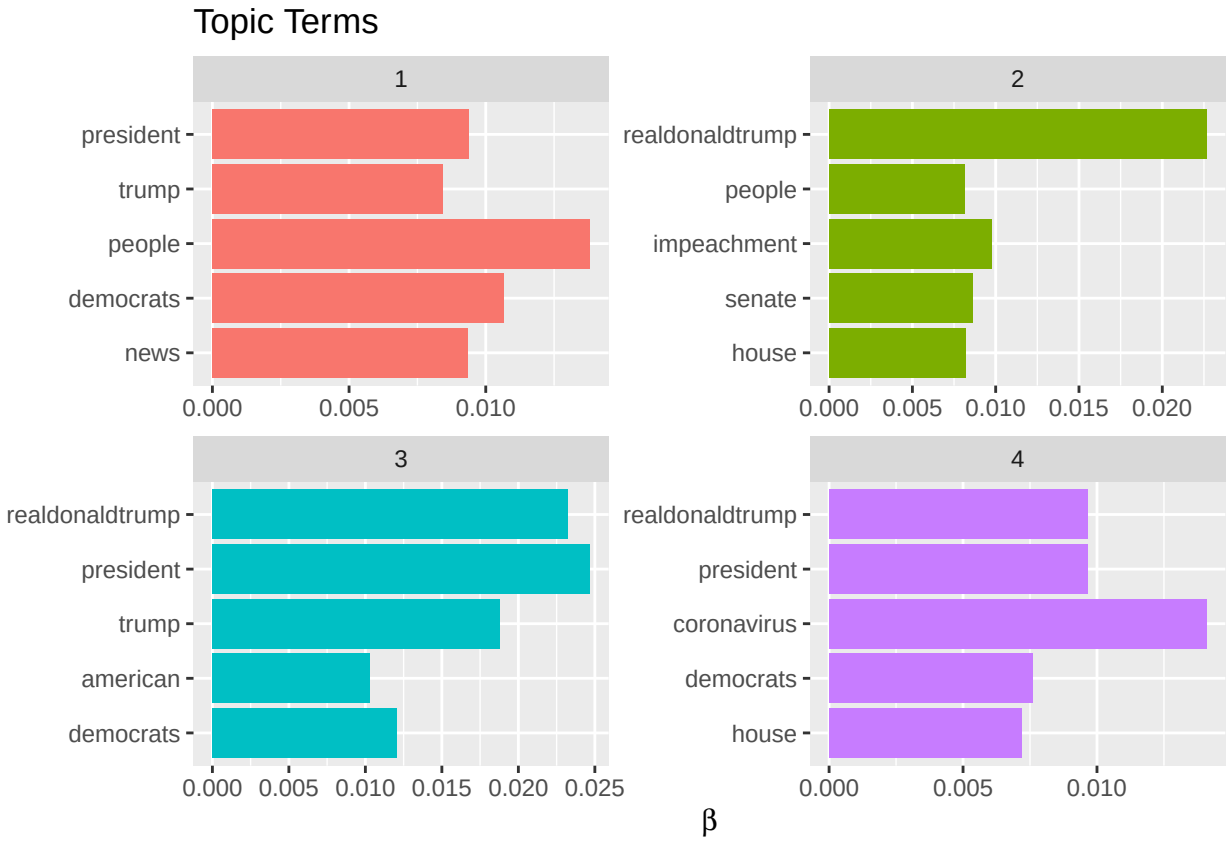
```
## A LDA_VEM topic model with 4 topics.
```

Topic Model

1. document term matrix needs to be cleaned up and generate beta use tidy
2. generate topic terms by extracting top_n 5 terms by beta that is group by topic
3. plot your topics use facet wrap and scales free.

Answer what topics did you identify? is there garbage in your topics, if so go back to the topic prep step and remove the junk and repeat.

```
tidy_lda <- tidy(tweet_lda, matrix = "beta")

topic_terms <- tidy_lda %>%
  ungroup() %>%
  group_by(topic) %>%
  top_n(5, beta) %>%
  arrange(topic,-beta)

topic_terms %>%
  ggplot(aes(reorder(term, beta), beta, fill = as.factor(topic))) +
  geom_col(show.legend = FALSE) +
  coord_flip() +
  labs(title = "Topic Terms", x = NULL, y = expression(beta)) +
  facet_wrap(~topic, ncol = 2, scales = "free")
```




```
# I identified the words "realdonaldtrump", "president" and "great" in the 4 topics.
```

Finally,

Based on your analysis of President Trump's tweets, what stood out to you? what did you think about this type of analysis. Write up your thoughts on this analysis.

I noticed that Trump mentions his name quite often on twitter. Other high-frequency terms are: "president", "democrats", "impeachment", "coronavirus", and "great". As for bigrams, he often mentions "fake news", "news media", and "joe biden".