A Study of Negative Mood on Reddit During Omicron Outbreak

Survmeth 727 Term paper

Xinran Wang, Yang Yu

2021-12-16

Abstract

This project investigates the effect of new variant - Omicron on people's mood and intend to figure out the proportions of different kinds of emotions, the changes of moods over time, and the relationship between negative moods and cases of Omicron variant in the US. Approximately 85,0000 comments in Reddit that is omicron-related were collected as our data. Then, we calculated the word counts and word frequency that were associated with negative emotion. We found that among eight negative emotions, fear was the dominant one across the time. Negative moods reached the peak on December 1. We didn't find strong correlation between negative moods and Covid-19 cases.

github: https://github.com/Xinranwxrtzxy/727-Mood-During-Omicron

This project needs the following libraries:

```
library(widyr)
```

```
## Warning: package 'widyr' was built under R version 4.1.2
```

```
library(stringr)
library(RedditExtractoR)
library(tidytext)
```

```
## Warning: package 'tidytext' was built under R version 4.1.2
```

```
library(rjson)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(knitr)
library(readxl)
```

Introduction

The Covid-19 epidemic has already engulfed the world for two years. From the initial global panic to the normalization of epidemic prevention, people are gradually accustomed to living with policies under the special period. However, new variants of the coronavirus strain are continuously emerging and they are much more contagious and threatening. "Omicron"- The World Health Organization named the latest variant on Nov 26th, and the first case of Omicron variant has been identified on Dec 1st. Cornell University and some universities in New York have locked down their campuses while switching the final exams online. Almost all people infected with the new variant are fully vaccinated. It did cause an uproar among the public and social media. According to Edmond Pui. et al (2020), abrupt changes to daily life are risk factors that can substantially affect people's mental health.

Hence, we assume people may experience a negative mood during the breakout of the Omicron. Since negative moods fall into many categories, eg. depression, anger, fear, sadness. We came up with the first research question as follow:

1.What kind of negative mood do people trigger most when the new variant - Omicron break-out?

Since we know two key time points of the Omicron and we intend to figure out whether people's negative emotions changed or not during different epidemic periods, the second research question is as followed:

2.Find the change of number and frequency of negative words during 'regular Covid-19 period' and 'key dates for Omicron'.

The subsequent report is structured in four key parts. The first part describes how we collect,clean and process data. The second part focuses on data analysis and visualization,while the last part details the results and our limitations. All data used are scraped from Reddit and are analyzed by R.

1. Gathering Data from Reddit

We gather data from Reddit using the collection tools for extracting structured data from reddit, `RedditExtractoR` version 3.0.5. It doesn't need key or token to get access to data. Firstly, we collect Reddit threads containing keyword "omicron" and their urls from the last month.

```
#links_omicron <- find_thread_urls(keywords = "omicron",
#                                   sort_by = "comments",
#                                   period = "month")
```

Then to get comments of each thread, we use the get_thread_content function. The result contain separate release dates of threads and comments, content of threads and comments, and other information that we do not need in this study.

```
#omicron <- get_thread_content(links_omicron$url)
#head(omicron)
```

Because we focus on analyzing the content of comments, we save the content and dates of comments as a separate object.

```
#comments_omicron<-omicron$comments
```

To process and analyze data without gathering data everytime, we parse the comments_omicron, which is a data frame object to the json object and then save as a local file.

```
#JSON_omicron<-toJSON(comments_omicron)
#write(JSON_omicron, file="D:/SurvMeth 727 Fundamentals of Computing and Data Display/all_comments_outp
```

Once the json file is stored on local computer, anyone can load json file without access to reddit data on their website. Since the object is character, before moving to data cleaning, we should convert it to a dataframe.

```
comments_omicron<-fromJSON(
  file="D:/SurvMeth 727 Fundamentals of Computing and Data Display/all_comments_output.json")
comments_omicron<-as.data.frame(comments_omicron)
```

Reddit Data Cleaning

We have columns and observations. Each row contain one comment. The names of columns are clear and easy to understand. Some variables like "votes" and "scores" are irrelavant to our research goals and we only want the dates, url and content of comments. This step helps us select the useful variables and discard the irrelevant ones. Besides, since On 26 November 2021, WHO designated the variant B.1.1.529 a variant of concern, named Omicron, the comments containing "omicron" before Nov 26 are definately not talking about omicron variant. So, we only filter the comments after 2021-11-25. Finally, we get the dataframe with three variables, date of comment, url, and comment content.

```
#select useful variables
df<-select(comments_omicron,c("date","comment","url"))
#give each comment a unique number
df<-df[order(df$date),]
df$no<-1:length(df[,1])
```

```
df<-as_tibble(df)
df$date <- ymd(df$date)
#delete comments not relevant to omicron variant
df<-filter(df,date>"2021-11-25")
head(df)
```

```
## # A tibble: 6 x 4
##   date       comment                          url                              no
##   <date>     <chr>                            <chr>                         <int>
## 1 2021-11-26 "Too late, already took a seco~ https://www.reddit.com/r/col~   205
## 2 2021-11-26 "The spike changes are extreme~ https://www.reddit.com/r/col~   206
## 3 2021-11-26 "Whatever, I\u0019m still gonn~ https://www.reddit.com/r/col~   207
## 4 2021-11-26 "Freak out has commenced"        https://www.reddit.com/r/col~   208
## 5 2021-11-26 "We pretty much failed to live~ https://www.reddit.com/r/col~   209
## 6 2021-11-26 "I'm not sure I buy that, hone~ https://www.reddit.com/r/col~   210
```

Reddit Data Exploration

After cleaning the original reddit data, we want to explore the top words in these comments collected in previous steps.

First of all, we split the comment into words by using unnest_tokens function from `tidytext` package. Then, we delete stopwords using a stopword lexicon from the `tidytext` package, also filter words containing at least one alphabetic, and delete numbers. Since some words repeatedly appear in one comment, we only keep the unique word in a comment. We get the word list containing 1099401 unique word of different comments. Each row containing one word of a comment and the number indicate which comment each word comes from.

```
df1<-df %>%
  unnest_tokens(output=word, input=comment) %>%
  anti_join(stop_words,by="word") %>%
  filter(str_detect(word,"[:alpha:]")) %>% #keep word containing at least one alpha
  filter(!str_detect(word,".*[0-9].*")) %>%  #delete numbers
  distinct() #if one word repeatedly appears in one comment, only count it once

head(df1)
```

```
## # A tibble: 6 x 4
##   date       url                                                    no word
##   <date>     <chr>                                               <int> <chr>
## 1 2021-11-26 https://www.reddit.com/r/collapse/comments/r2z3ul/we_~   205 late
## 2 2021-11-26 https://www.reddit.com/r/collapse/comments/r2z3ul/we_~   205 mortg~
## 3 2021-11-26 https://www.reddit.com/r/collapse/comments/r2z3ul/we_~   205 buy
## 4 2021-11-26 https://www.reddit.com/r/collapse/comments/r2z3ul/we_~   205 toilet
## 5 2021-11-26 https://www.reddit.com/r/collapse/comments/r2z3ul/we_~   205 paper
## 6 2021-11-26 https://www.reddit.com/r/collapse/comments/r2z3ul/we_~   205 zm
```

Then we create a top words list to glance at the words with the highest frequency of occurrence. We only look at the words occurred more than 1000 times in all comments.

```
comment_word<-df1 %>%
  ungroup() %>%
  count(word,name="n",sort = T) %>%
  filter(n>=500)
head(comment_word,100)
```

```
## # A tibble: 100 x 2
##    word           n
##    <chr>      <int>
##  1 people     15705
##  2 covid      10582
##  3 time        6366
##  4 omicron     6077
##  5 vaccine     5446
##  6 vaccinated  5323
##  7 variant     4750
##  8 gt          4145
##  9 virus       3803
## 10 https       3666
## # ... with 90 more rows
```

From the top words list, we found four words that are related to negative moods are mentioned over 1000 times. They are "bad", "shit", "fuck", and "die". We choose them as the indication of negative moods and have a look at the word count of these four words.

```
comment_word %>%
  subset(word %in% c("bad", "shit", "fuck","die"))
```

```
## # A tibble: 4 x 2
##   word      n
##   <chr> <int>
## 1 bad    2429
## 2 shit   2159
## 3 fuck   1621
## 4 die    1424
```

After deciding on the word we will use to analyze, we calculate the overall word frequency (word count/comment count per day) and save the data as df2. We also calculate the different word count of four words per day and save it as df3. Then we merge df2 and df3 by using inner_join function and calculate the different word frequency per day by dividing separate word count by overall word count. To make the name of column more straightforward, the total.day represent the overall word count and the total.word represent the separate word count per day.

```
#the overall word count per day
df2<-df1 %>%
  group_by(date) %>%
  summarise(total = n())
#the word count of four words per day
df3<-df1 %>%
  filter(word %in% c("bad", "shit", "fuck","die")) %>%
  group_by(date,word) %>%
  summarise(total = n())
```

```
## `summarise()` has grouped output by 'date'. You can override using the `.groups` argument.
```

```
df4<-inner_join(df2,df3,by="date")
#the word frequency per day (/percent)
df4$frequency<-df4$total.y/df4$total.x*100
df4<-rename(df4,total.day=total.x,total.word=total.y)
```

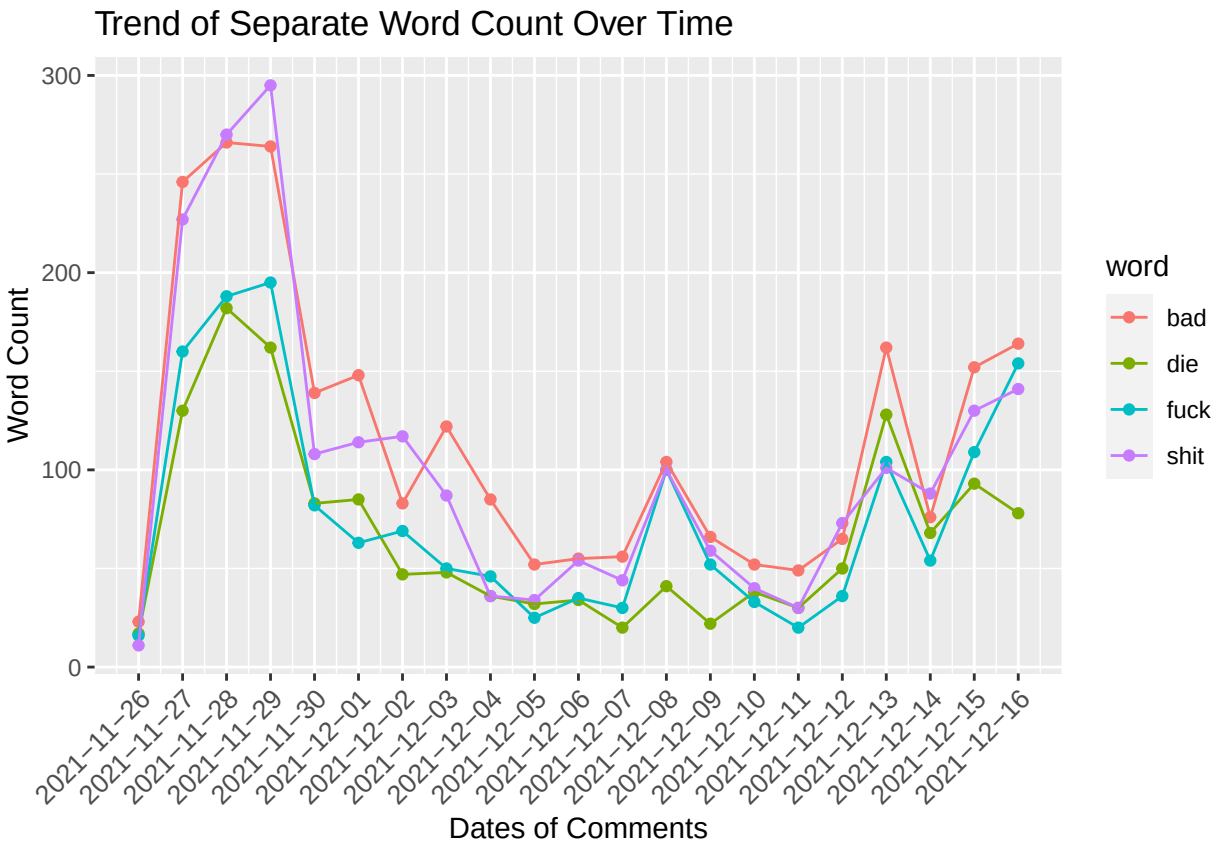
Now we can plot the trend of separate word count, overall comments count, separate word frequency, and frequency of sum of four words over time.

```
plot1<-df1 %>%
  filter(word %in% c("bad", "shit", "fuck","die")) %>%
  group_by(date,word) %>%
  summarise(total = n()) %>%
  ggplot(aes(date,total,color = word,group = word)) +
  geom_point()+
  geom_line()+
  scale_x_date(breaks = unique(df1$date)) +
  labs(title="Trend of Separate Word Count Over Time",
       x="Dates of Comments",
       y="Word Count")+
  theme(axis.text.x = element_text(angle = 45, size = 10,hjust = 1))
```

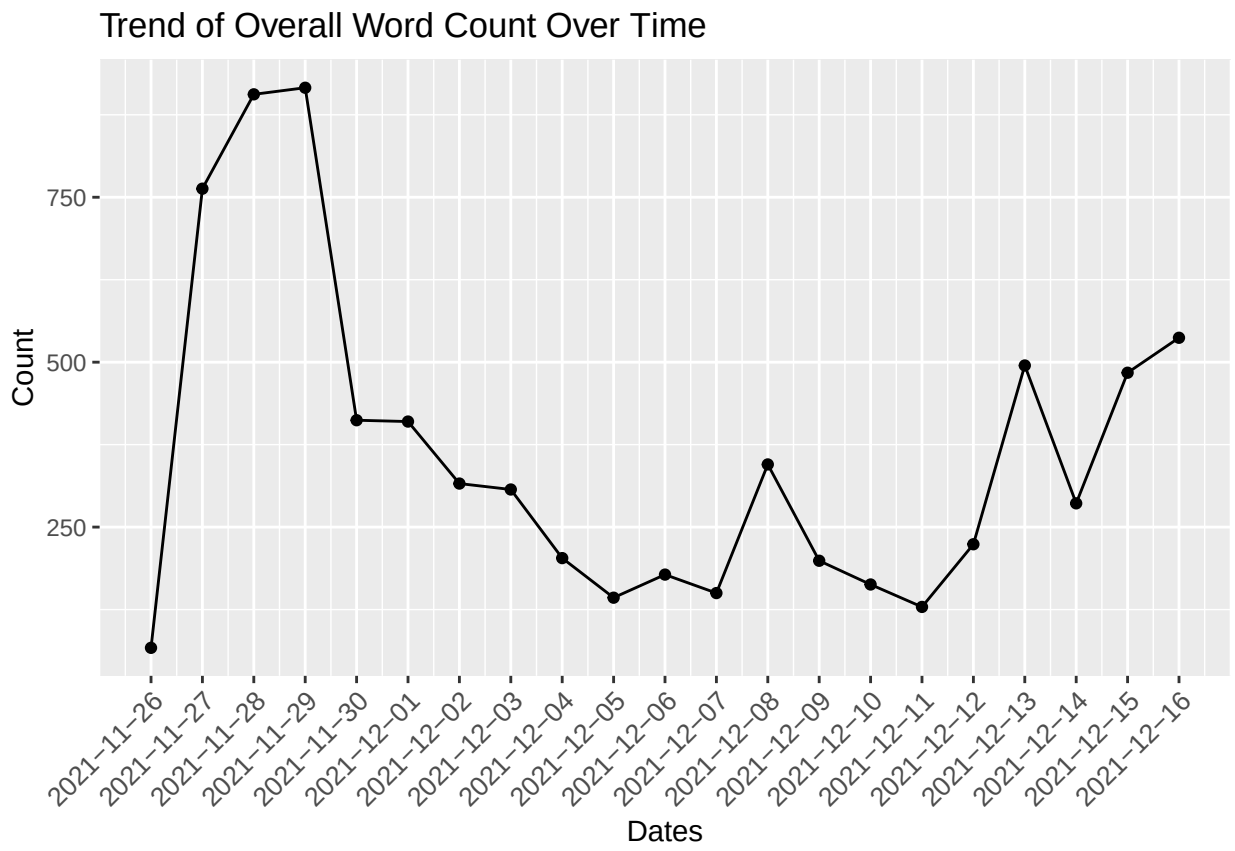
```
## `summarise()` has grouped output by 'date'. You can override using the `.groups` argument.
```

```
plot2<-df1 %>%
    filter(word %in% c("bad", "shit", "fuck","die")) %>%
  group_by(date) %>%
  summarise(total = n()) %>%
  ggplot(aes(date,total)) +
  geom_point()+
  geom_line()+
  scale_x_date(breaks = unique(df1$date)) +
  labs(title="Trend of Overall Word Count Over Time",
       x="Dates",
       y="Count")+
  theme(axis.text.x = element_text(angle = 45, size = 10,hjust = 1))

plot3<-df4 %>%
  filter(word %in% c("bad", "shit", "fuck","die")) %>%
  group_by(date,word) %>%
  ggplot(aes(date,frequency,color = word,group = word)) +
  geom_point()+
  geom_line()+
  scale_x_date(breaks = unique(df4$date)) +
  labs(title="Trend of Separate Word Frequency Over Time",
       x="Dates of Comments",
       y="Word Frequency")+
  theme(axis.text.x = element_text(angle = 45, size = 10,hjust = 1))

plot4<-df4 %>%
  filter(word %in% c("bad", "shit", "fuck","die")) %>%
```

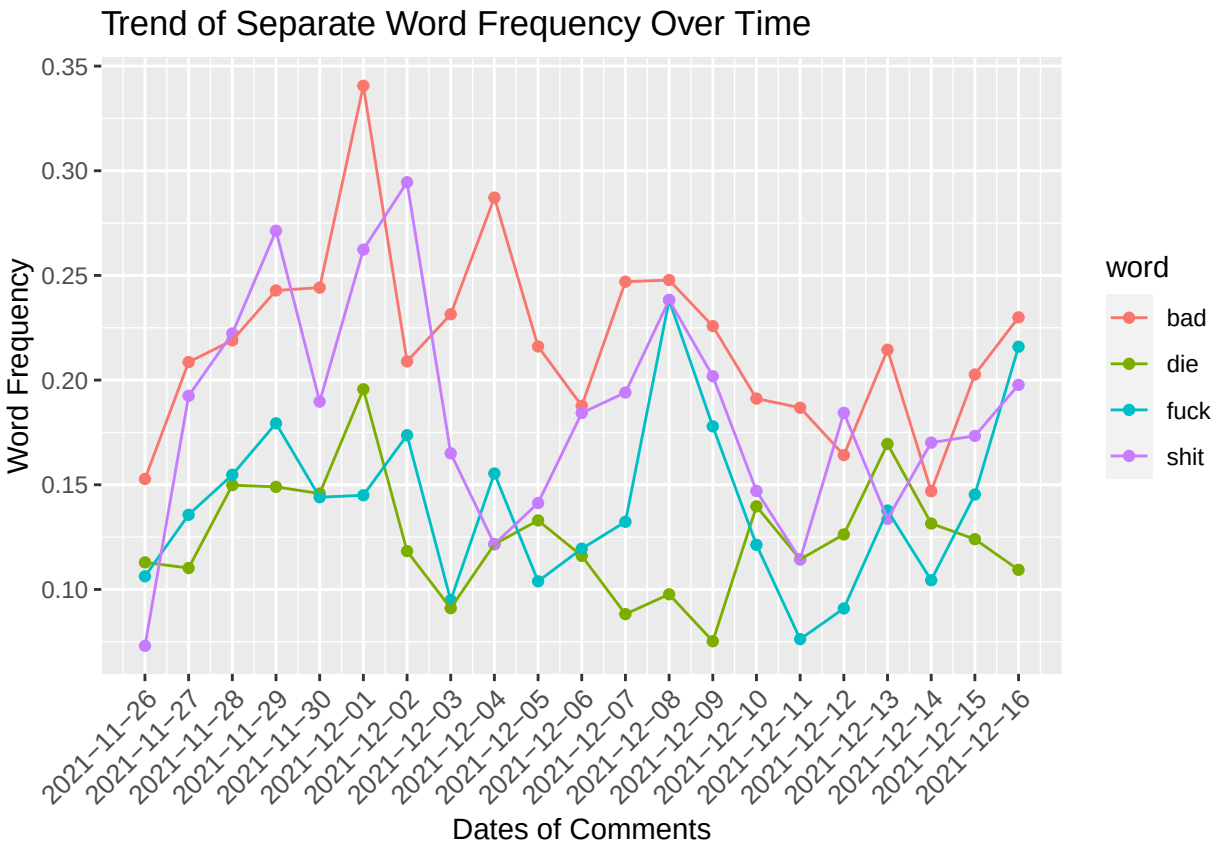
```
  group_by(date) %>%
  summarise(total = sum(frequency)) %>%
  ggplot(aes(date,total)) +
  geom_point()+
  geom_line()+
  scale_x_date(breaks = unique(df4$date)) +
  labs(title="Trend of Overall Word Frequency Over Time",
       x="Dates of Comments",
       y="Word Frequency")+
  theme(axis.text.x = element_text(angle = 45, size = 10,hjust = 1))
plot1
```



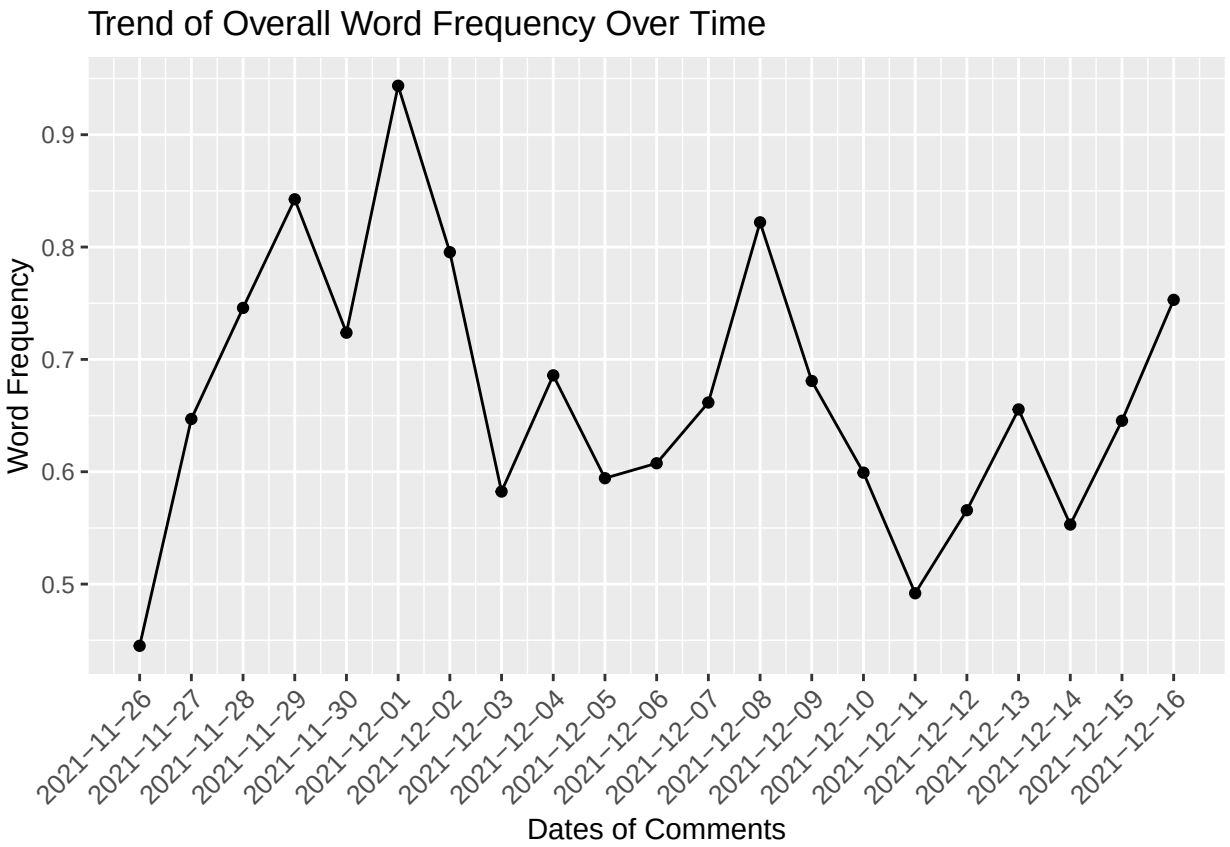

```
plot2
```


Trend of Overall Word Count Over Time
Count
750
500
250
2021−11−26
2021−11−27
2021−11−28
2021−11−29
2021−11−30
2021−12−01
2021−12−02
2021−12−03
2021−12−04
2021−12−05
2021−12−06
2021−12−07
2021−12−08
2021−12−09
2021−12−10
2021−12−11
2021−12−12
2021−12−13
2021−12−14
2021−12−15
2021−12−16
Dates

```
plot3
```


Trend of Separate Word Frequency Over Time
Word Frequency
0.35
0.30
0.25
0.20
0.15
0.10
2021−11−26
2021−11−27
2021−11−28
2021−11−29
2021−11−30
2021−12−01
2021−12−02
2021−12−03
2021−12−04
2021−12−05
2021−12−06
2021−12−07
2021−12−08
2021−12−09
2021−12−10
2021−12−11
2021−12−12
2021−12−13
2021−12−14
2021−12−15
2021−12−16
Dates of Comments
word
bad
die
fuck
shit

```
plot4
```



#Reddit Data Exploration with NRC Word-Emotion Association Lexicon

Except for the four specific words, we think it would be better if we can include other words to analyzing the emotions. So we import the NRC Word-Emotion Association Lexicon to analyze the emotion, including anger, fear, anticipation, trust, surprise, sadness, joy, and disgust. (https://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm)

```
NRC<-read_excel("D:/SurvMeth 727 Fundamentals of Computing and Data Display/NRC-Emotion-Lexicon-v0.92-In
```

We merge the NRC data with comment word count data, so that we can easily see the the word, word count and the emotion associated with it. There are eight emotions associated with each word, the number in emotion variable indicate the score. Number "1" indicate the word is associated with this specific emotion and number "0" indicates no association.

```
tempx<-comment_word$word[match(NRC$word,comment_word$word)]
tempx<-tempx %>%
  na.omit() %>%
  as_tibble() %>%
  rename(word=value)
tempx<-merge(tempx,comment_word,by="word") %>%
  merge(NRC,by="word")
tempx<-tempx[order(tempx$n,decreasing=T),]
#delete neutral words, delete positive
word_mood<-tempx[rowSums(tempx[,3:12])>0,]
head(word_mood)
```

```
##         word    n Positive Negative Anger Anticipation Disgust Fear Joy Sadness
```

```
## 205     time 6366        0        0     0            1       0    0    0       0
## 212  vaccine 5446        1        0     0            0       0    0    0       0
## 213    virus 3803        0        1     0            0       0    0    0       0
## 54       don 3478        1        0     0            0       0    0    0       0
## 141 pandemic 2596        0        1     0            0       0    1    0       1
## 11       bad 2429        0        1     1            0       1    1    0       1
##     Surprise Trust
## 205        0     0
## 212        0     0
## 213        0     0
## 54         0     1
## 141        0     0
## 11         0     0
```

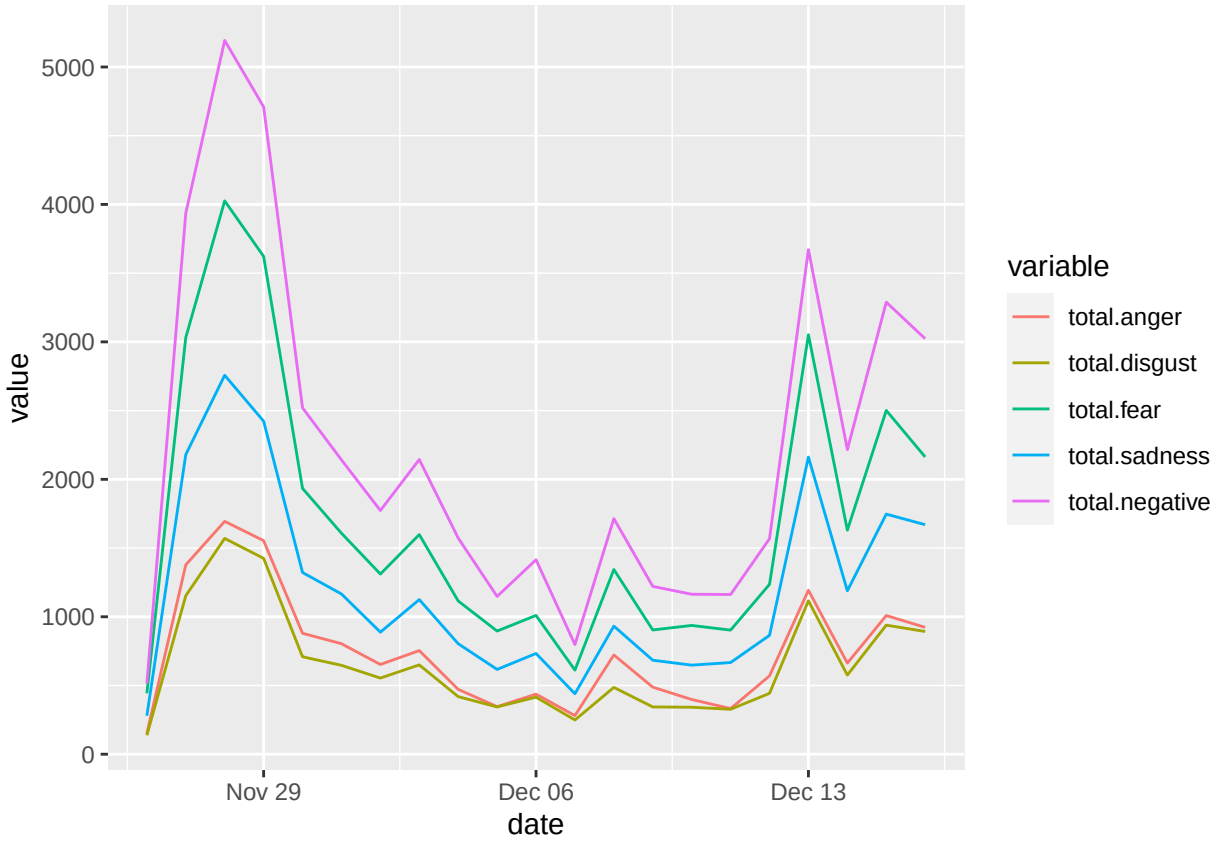
```
#merge the neutral word-emotion association with original data
emo1<-merge(df1,word_mood,by="word")
```

Since we don't want emotion score at word-level, we need to calculate the total scores on different emotions, grouping by date. Higher the score, stronger the emotion associated with the word. Besides total scores, the relative proportion of each emotion everyday is also calculated.

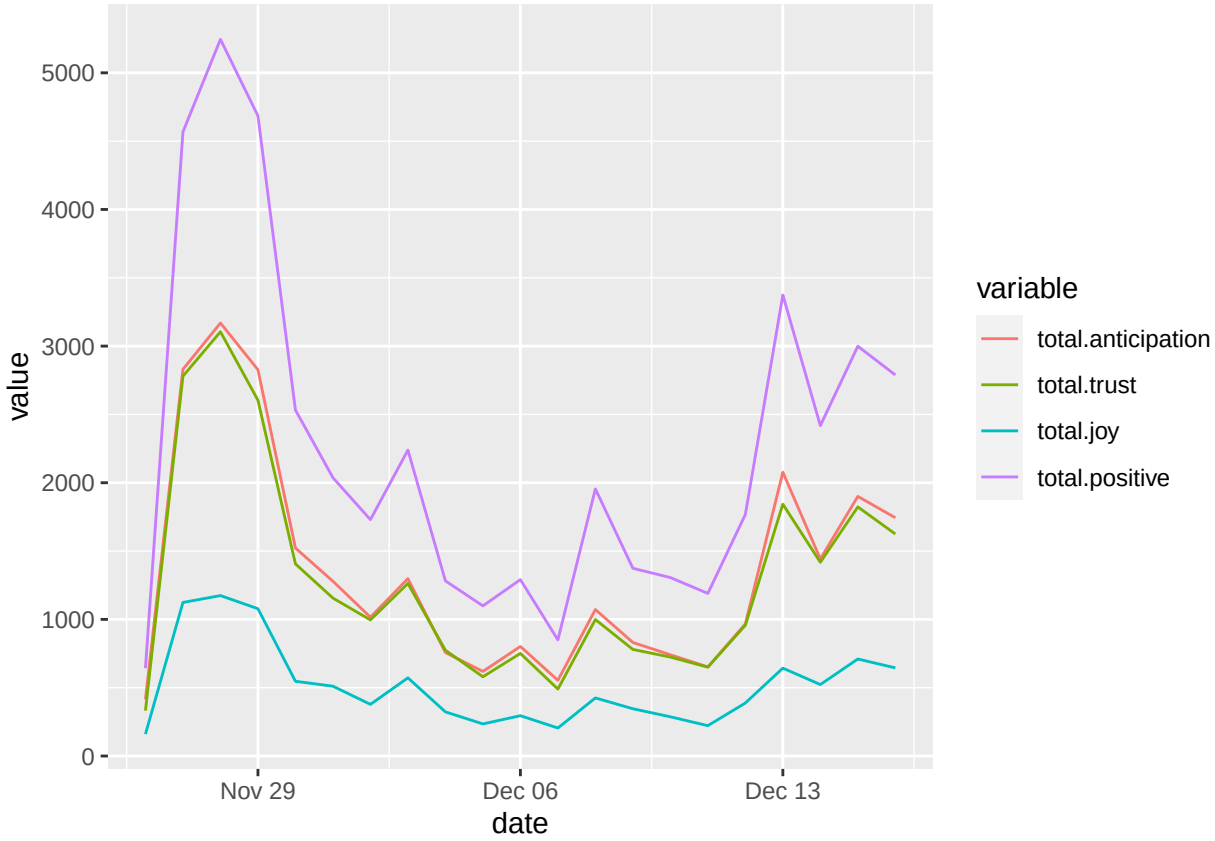
```
#calculate the overall scores for emotions each day
emo2<-emo1%>%
  group_by(date) %>%
  summarise(total.anger=sum(Anger),
            total.anticipation=sum(Anticipation),
            total.disgust=sum(Disgust),
            total.fear=sum(Fear),
            total.sadness=sum(Sadness),
            total.joy=sum(Joy),
            total.trust=sum(Trust),
            total.negative=sum(Negative),
            total.positive=sum(Positive)
            )
#calculate the proportion of specific emotion scores
#over all emotion scores each day
emo3<-emo2 %>%
  mutate(total=rowSums(.[2:8]))
emo4<-emo3%>%
  group_by(date) %>%
  summarise(freq.anger=(total.anger/total)*100,
            freq.anticipation=(total.anticipation/total)*100,
            freq.disgust=(total.disgust/total)*100,
            freq.fear=(total.fear/total)*100,
            freq.sadness=(total.sadness/total)*100,
            freq.joy=(total.joy/total)*100,
            freq.trust=(total.trust/total)*100,
            freq.negative=(total.negative/total)*100,
            freq.positive=(total.positive/total)*100
            )
```

After processing the data, we can plot to explore the relationship between different emotions and time as well as the proportion of different emotions.

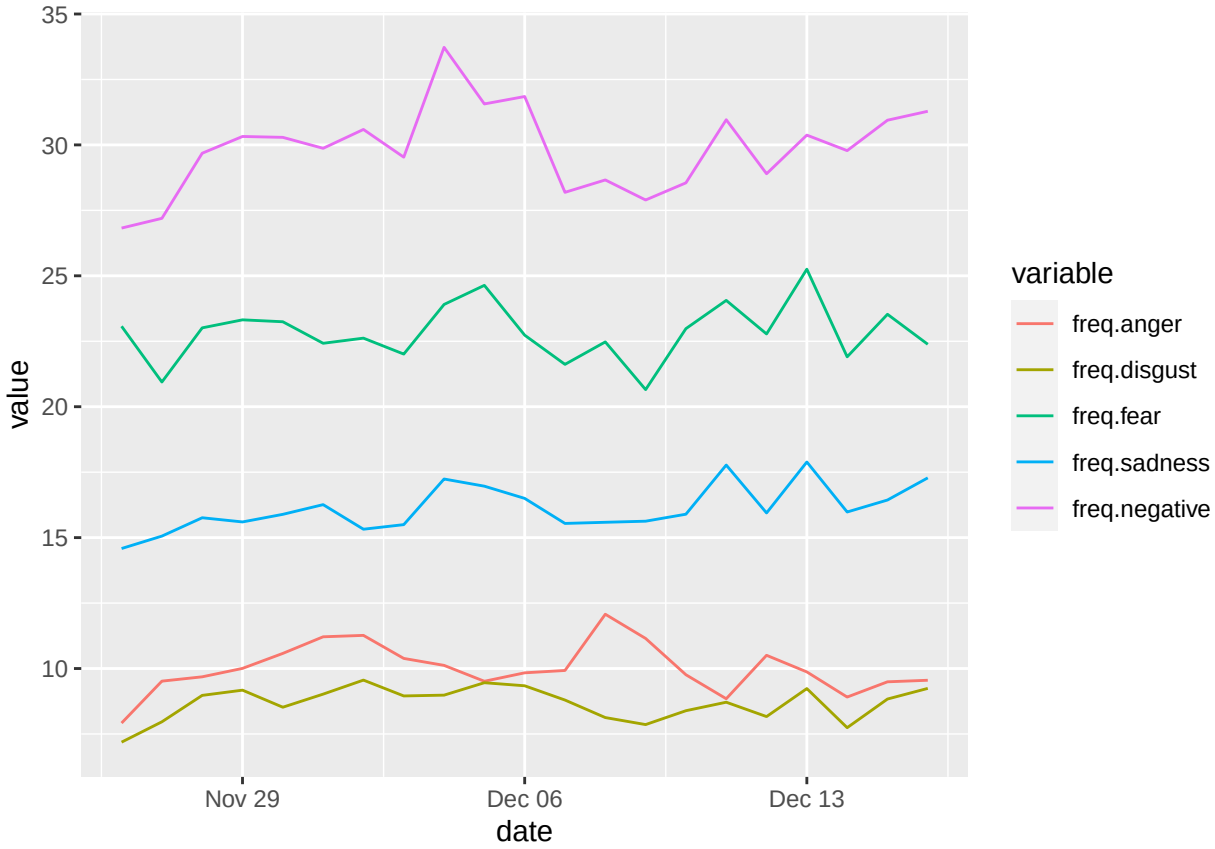
```
library(reshape2)
#scores
emo.neg <- melt(emo2, id = "date",
                measure = c("total.anger", "total.disgust",
                            "total.fear","total.sadness",
                            "total.negative"))
(plotemo1<-ggplot(emo.neg,aes(date,value,colour=variable)) +
    geom_line())
```



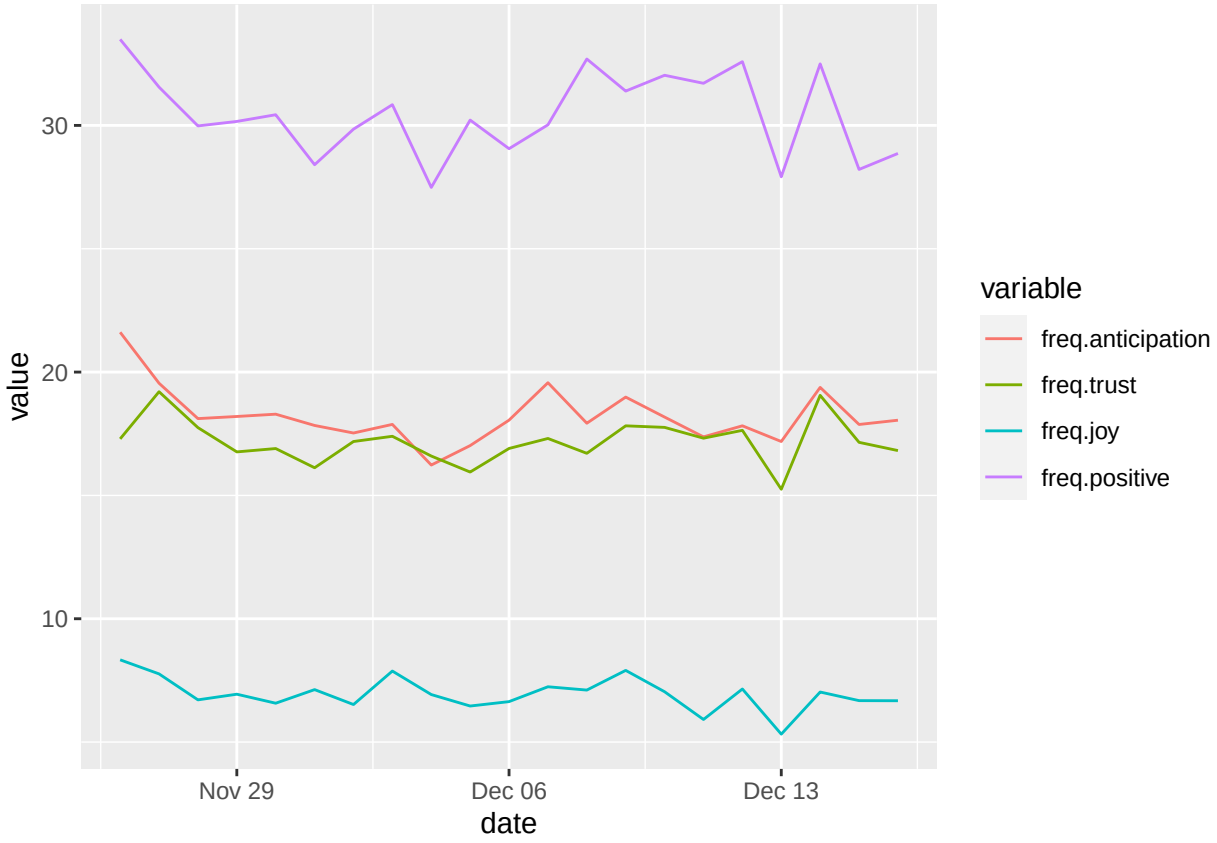

```
emo.pos <- melt(emo2, id = "date",
                measure = c("total.anticipation", "total.trust",
                            "total.joy","total.positive"))
(plotemo2<-ggplot(emo.pos,aes(date,value,colour=variable)) +
    geom_line())
```




```
#proportion of scores
emo.neg <- melt(emo4, id = "date",
                measure = c("freq.anger", "freq.disgust",
                            "freq.fear","freq.sadness",
                            "freq.negative"))
(plotemo1<-ggplot(emo.neg,aes(date,value,colour=variable)) +
    geom_line())
```




```
emo.pos <- melt(emo4, id = "date",
                measure = c("freq.anticipation", "freq.trust",
                            "freq.joy","freq.positive"))
(plotemo2<-ggplot(emo.pos,aes(date,value,colour=variable)) +
    geom_line())
```



3. Gathering Data and Data Cleaning from CDC

We import data about Covid 19 cases in US (downloaded from CDC https://covid.cdc.gov/covid-data-tracker/#trends_dailycases). We originally plan to gather the data of Omicron variant cases, but we failed to find the reliable data resources. So we use the Covid-19 cases data from CDC instead.

```
covid<-read.csv(
  file="D:/SurvMeth 727 Fundamentals of Computing and Data Display/data_table_for_daily_case_trends__th
```

Before merging CDC data with Reddit data, we clean the CDC data to only include data after Nov 25, and select variables of dates and number of new Covid-19 cases per day.

```
names(covid)
```

```
## [1] "ï..State"          "Date"              "New.Cases"
## [4] "X7.Day.Moving.Avg" "Historic.Cases"
```

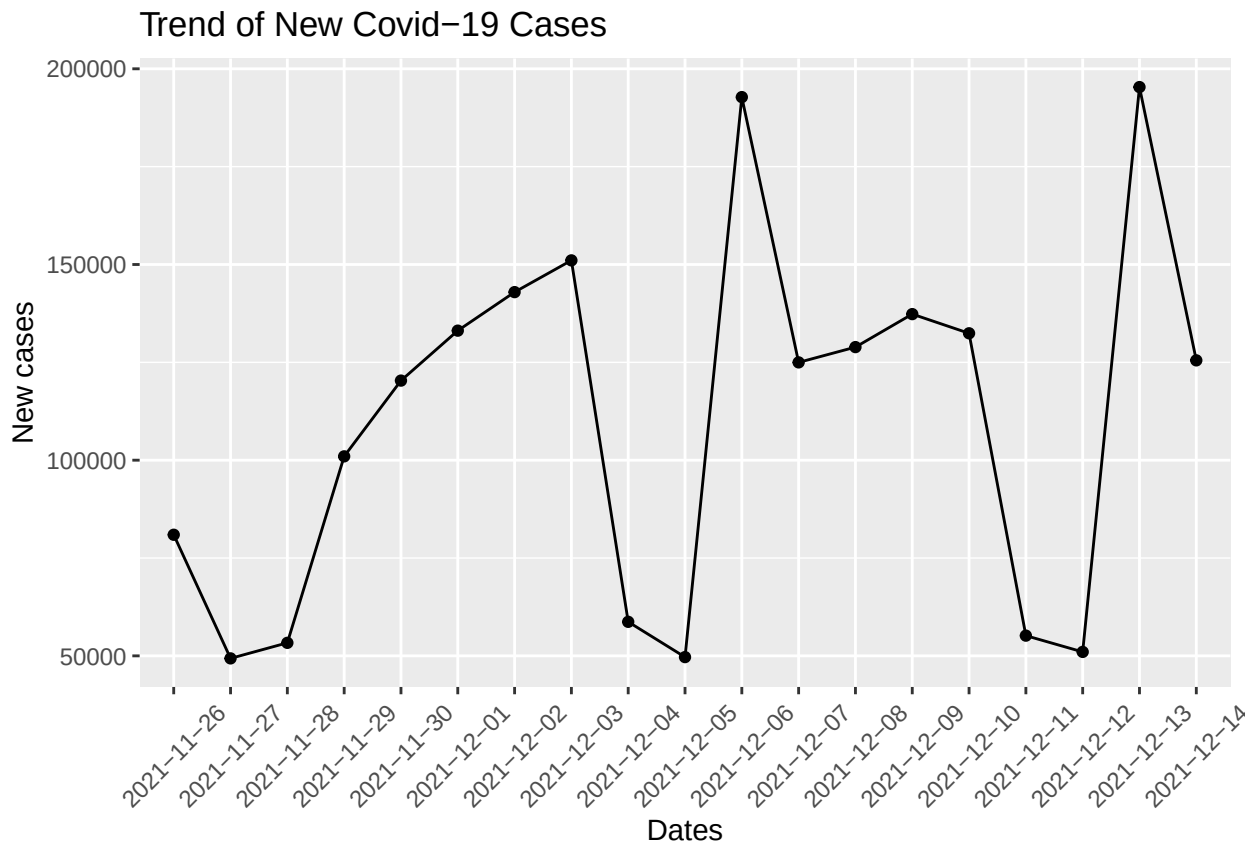
```
covid<-rename(covid,date=Date,X7.Days=X7.Day.Moving.Avg)
covid$date<-mdy(covid$date)
covid<-covid %>%
  filter(date>"2021-11-25") %>%
  select(date,New.Cases,X7.Days) %>%
  arrange(date)
head(covid)
```

```
##         date New.Cases X7.Days
## 1 2021-11-26     80947   83644
## 2 2021-11-27     49356   85128
## 3 2021-11-28     53334   87881
## 4 2021-11-29    100978   80288
## 5 2021-11-30    120332   82913
## 6 2021-12-01    133097   86893
```

#CDC Data Exploration

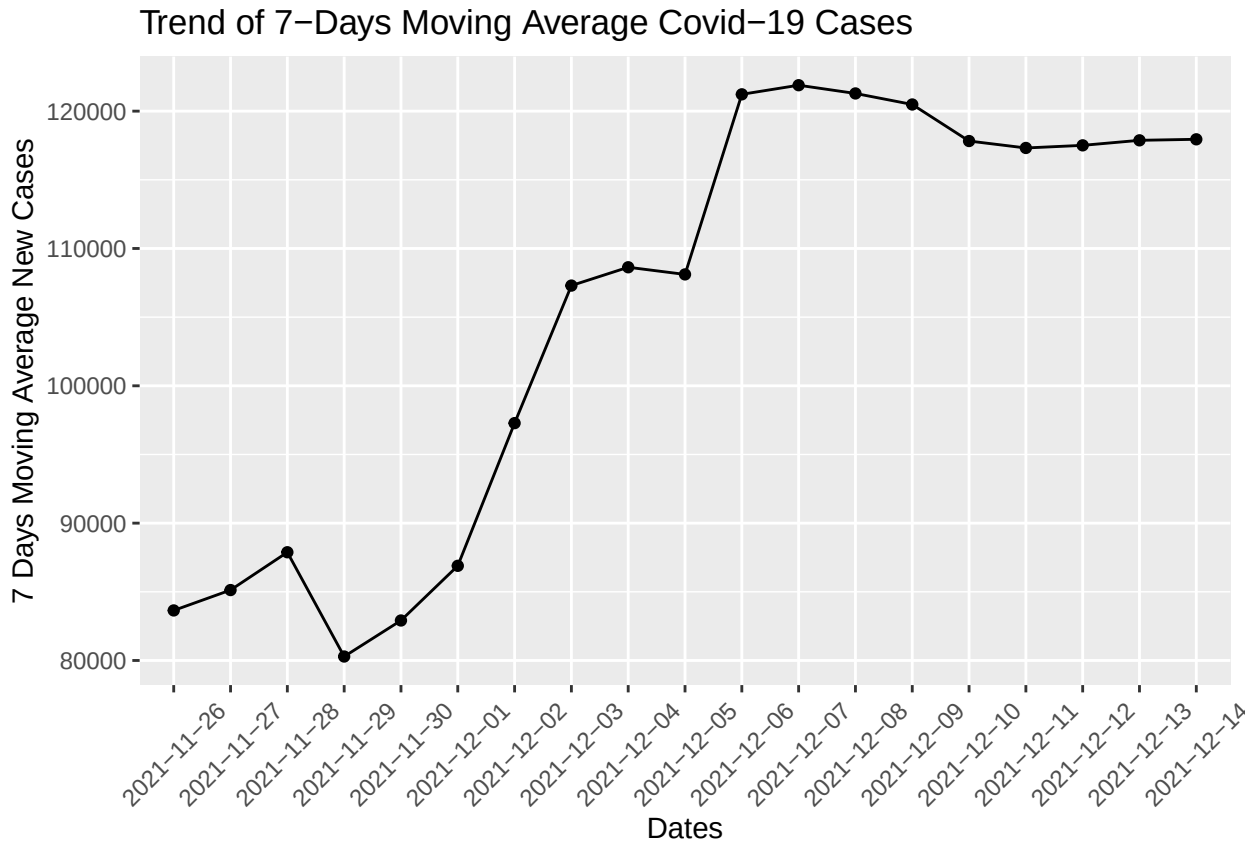
We explore the trend of Covid-19 new cases over time by plotting.

```
#New cases over time
covid %>%
  ggplot(aes(x=factor(date),y=New.Cases,group=1))+
  geom_line()+geom_point() +theme(axis.text.x = element_text(angle = 45, hjust = 0.5, vjust = .5)) +
  labs(title="Trend of New Covid-19 Cases",
       x="Dates",
       y="New cases")
```

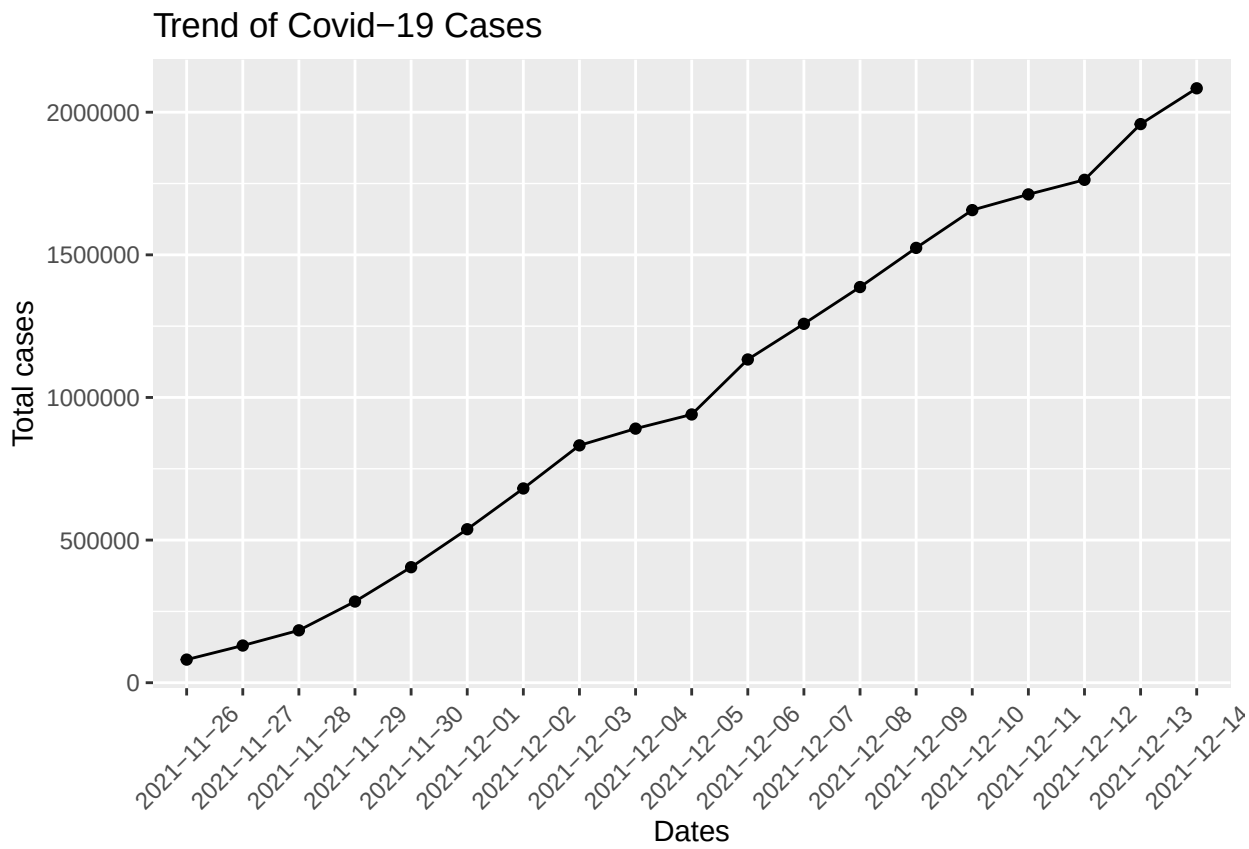



```
#7-Days Moving Average over time
covid %>%
  ggplot(aes(x=factor(date),y=X7.Days,group=1))+
  geom_line()+geom_point() +theme(axis.text.x = element_text(angle = 45, hjust = 0.5, vjust = .5)) +
  labs(title="Trend of 7-Days Moving Average Covid-19 Cases",
```

```
      x="Dates",
      y="7 Days Moving Average New Cases")
```




```
#Cumulative Cases over time
covid %>%
  mutate(total=cumsum(New.Cases)) %>%
  ggplot(aes(x=factor(date),y=total,group=1))+
  geom_line()+geom_point() +theme(axis.text.x = element_text(angle = 45, hjust = 0.5, vjust = .5)) +
  labs(title="Trend of Covid-19 Cases",
       x="Dates",
       y="Total cases")
```



Comparing these plot, we found 7-Days moving average data is more reasonable, compared to new cases per day, because of the low new cases in weekends due to less tests. Besides, the increase of 7-Days moving average become rapid after Nov 29, and increase speed tends to be steady after Dec 5.

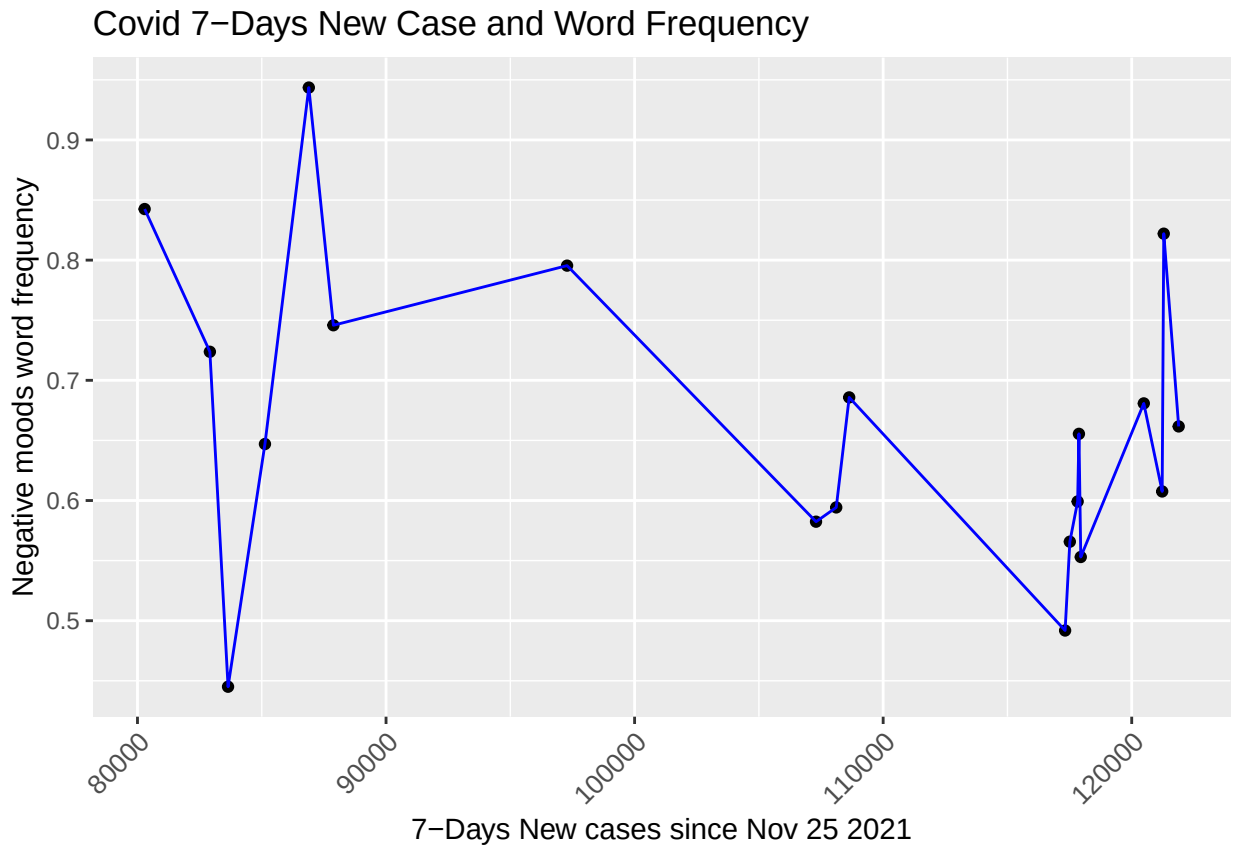
Then we can combine Covid 7-Days moving average cases and word frequency per day

```
df5<-df4 %>%
  group_by(date,word) %>%
  mutate(total = sum(frequency)) %>%
  inner_join(covid,by="date") %>%
  select(date,total,X7.Days,word,New.Cases)
head(df5)
```

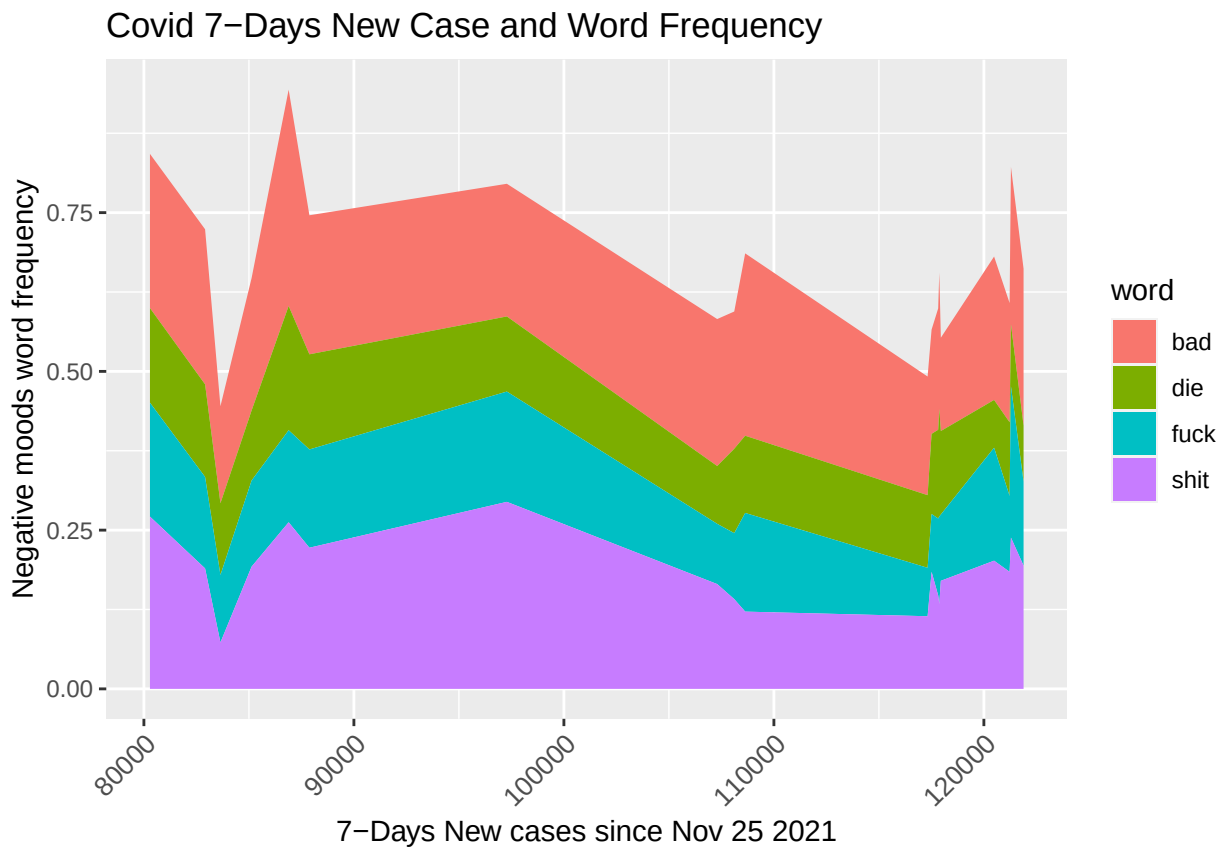
```
## # A tibble: 6 x 5
## # Groups:   date, word [6]
##   date        total X7.Days word  New.Cases
##   <date>      <dbl>   <int> <chr>     <int>
## 1 2021-11-26 0.153    83644 bad       80947
## 2 2021-11-26 0.113    83644 die       80947
## 3 2021-11-26 0.106    83644 fuck      80947
## 4 2021-11-26 0.0731   83644 shit      80947
## 5 2021-11-27 0.209    85128 bad       49356
## 6 2021-11-27 0.110    85128 die       49356
```

We also want to visualize the relationship between covid cases and word frequency while ignoring dates to see if there are any relationship.

```
df5 %>%
  group_by(X7.Days) %>%
  summarise(total=sum(total)) %>%
  ggplot() +
  geom_point(mapping=aes(X7.Days,total))+
  geom_line(mapping=aes(X7.Days,total),color = "blue")+
  labs(title="Covid 7-Days New Case and Word Frequency",
       x="7-Days New cases since Nov 25 2021",
       y="Negative moods word frequency")+
  theme(axis.text.x = element_text(angle = 45, size = 10,hjust = 1))
```




```
df5 %>%
  ggplot(mapping=aes(X7.Days,total,fill=word)) +
  geom_area()+
  labs(title="Covid 7-Days New Case and Word Frequency",
       x="7-Days New cases since Nov 25 2021",
       y="Negative moods word frequency")+
  theme(axis.text.x = element_text(angle = 45, size = 10,hjust = 1))
```



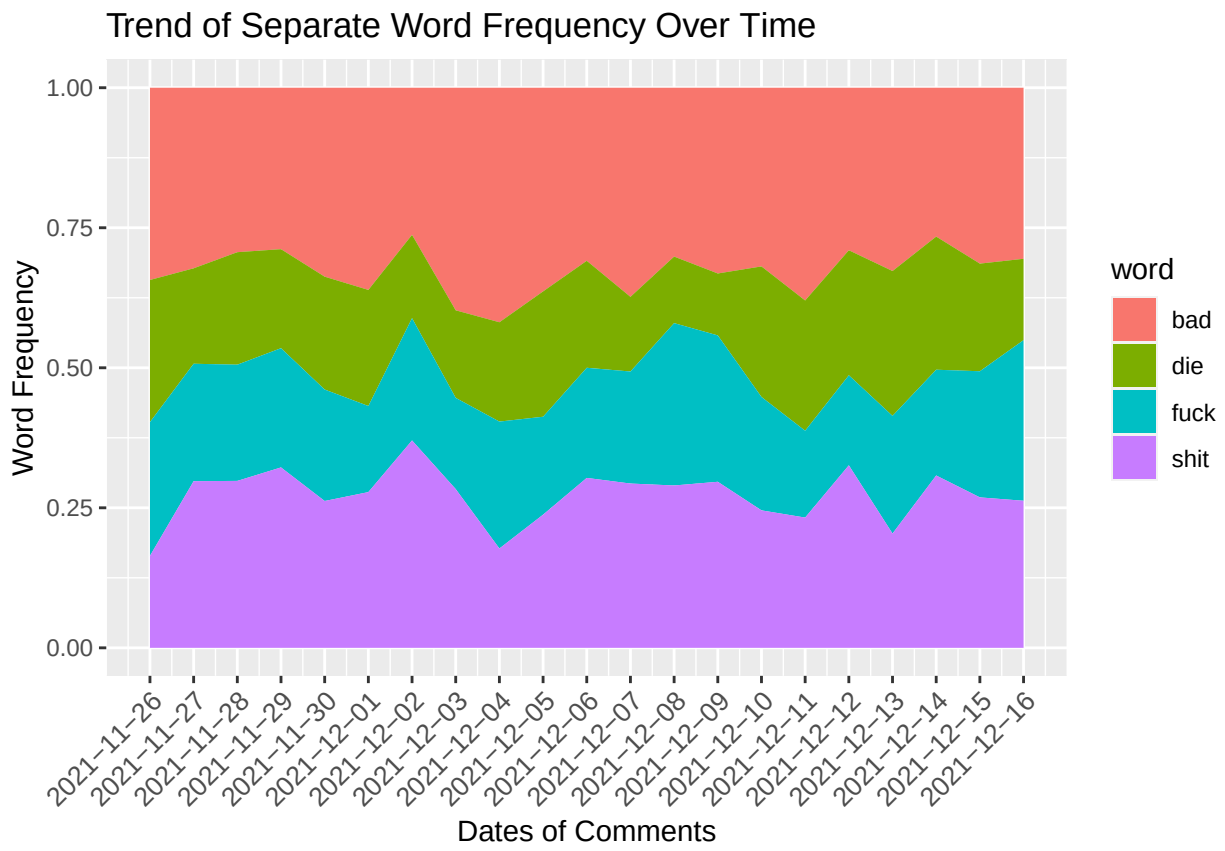
However, it seems that the relationship between the number of new cases and negative moods word is not clear.

##4. Result

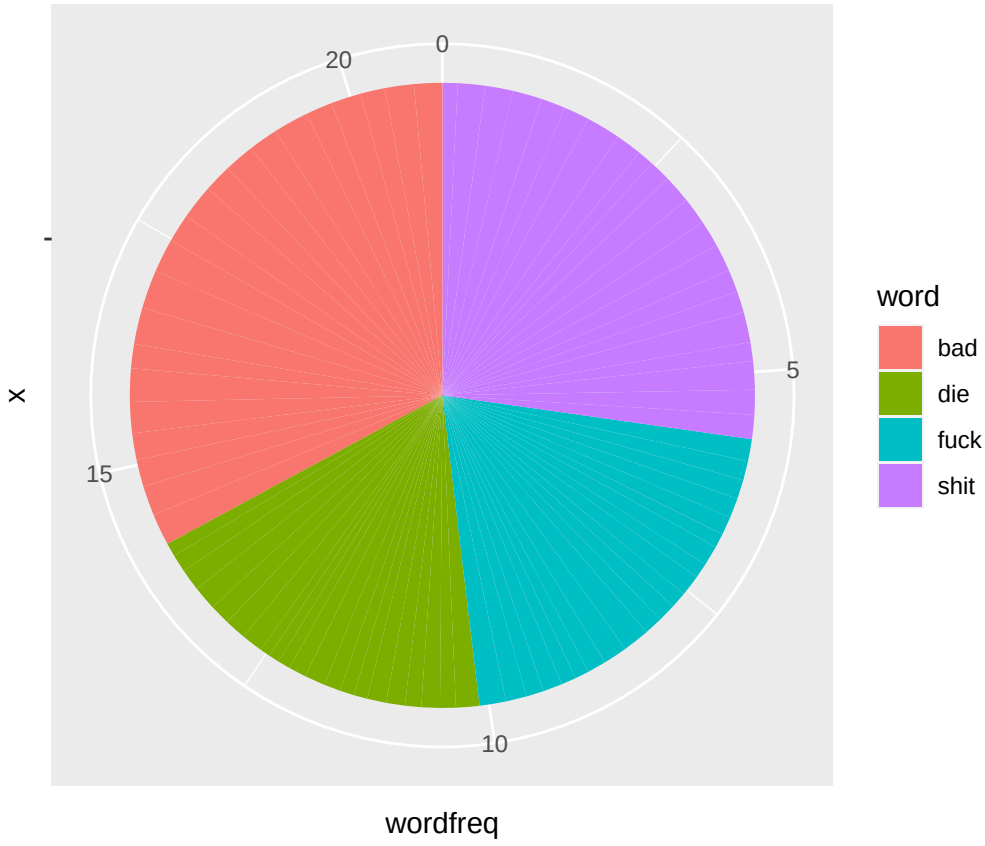
#Research question 1: What kind of negative mood do people trigger most after the new variant - Omicron breakout?

We plot the changes from Nov 26 to Dec 16 for word frequency of four words in total and separately.

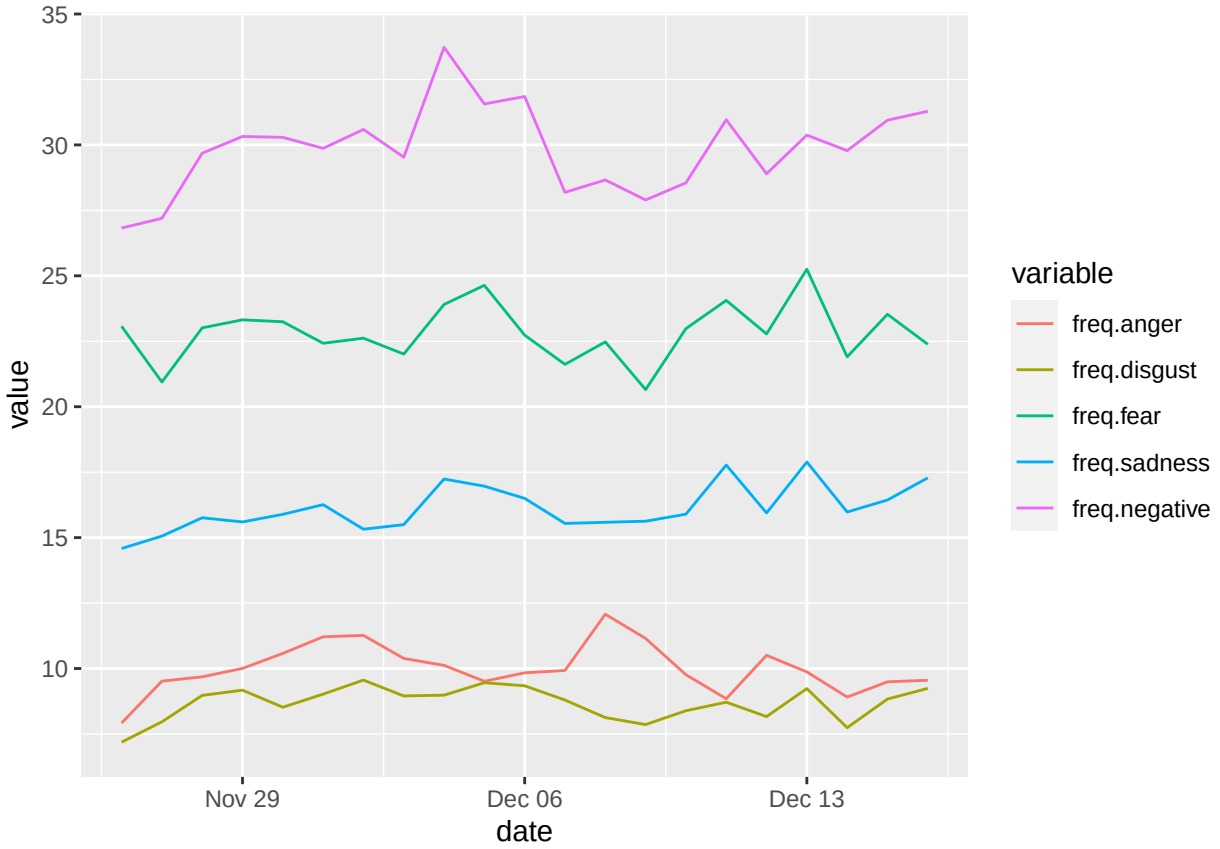
```
#calculate the word frequency as the number of comments containing one specific word / containing any o
df6<-df4 %>%
  group_by(date) %>%
  mutate(n=sum(total.word)) %>%
  mutate(wordfreq=total.word/n)
df6 %>%
  group_by(date,word) %>%
  ggplot() +
  geom_area(aes(date,wordfreq,fill=word))+
  scale_x_date(breaks = unique(df6$date)) +
  labs(title="Trend of Separate Word Frequency Over Time",
       x="Dates of Comments",
       y="Word Frequency")+
  theme(axis.text.x = element_text(angle = 45, size = 10,hjust = 1))
```




```
df6 %>%
  ggplot(aes(x="",y=wordfreq,fill=word))+
  geom_bar(stat="identity")+
  coord_polar(theta="y")
```




```
ggplot(emo.neg,aes(date,value,colour=variable)) +  geom_line()
```

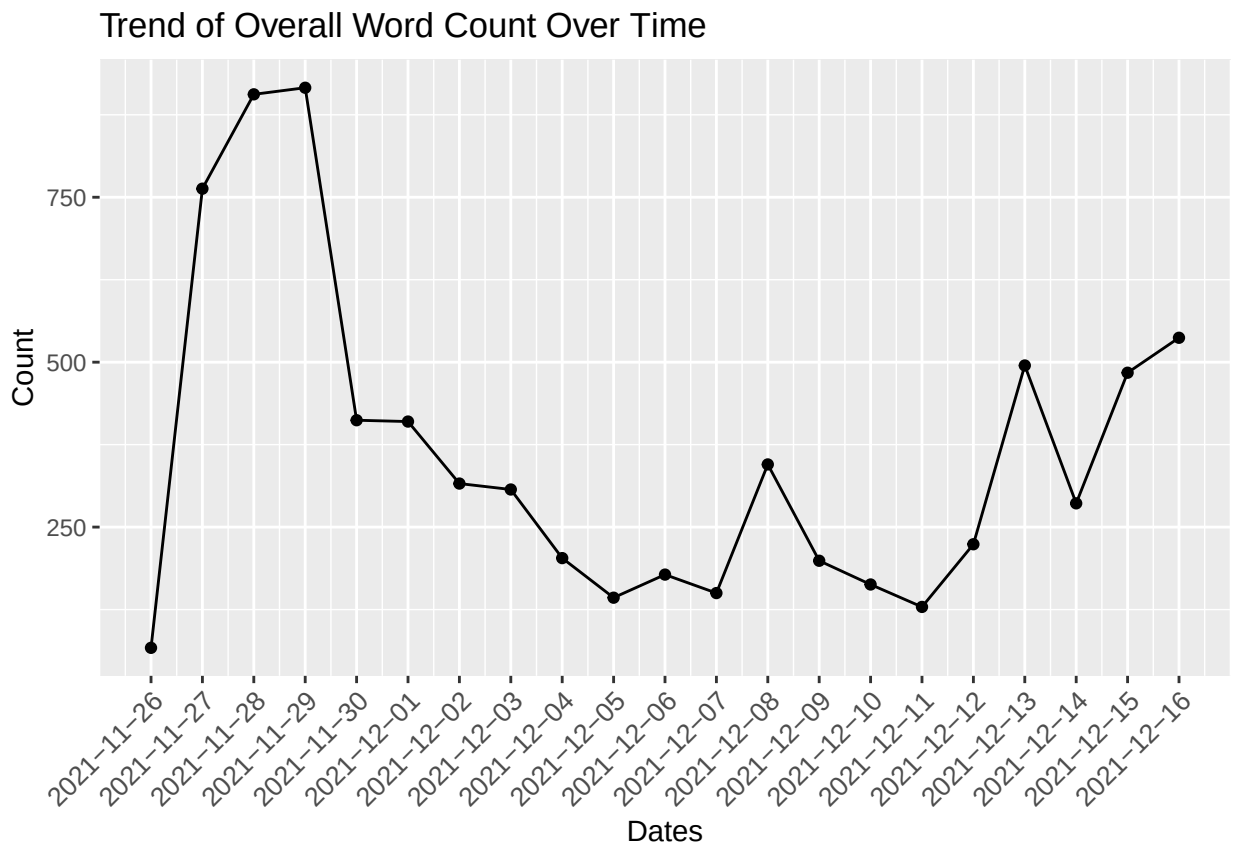


From the first two plots, we find that if we focus on the most frequent four words, "bad" and "shit" are always more popular than other two words, maybe indicating the anger emotion. However, to get a more convincing result, we should focus on the big-piture - the third plot, which shows the relative proportion of different negative emotions of all words in comments per day. Fear becomes the dominant mood for reddit users, followed by sadness, negativeness, and disgust.

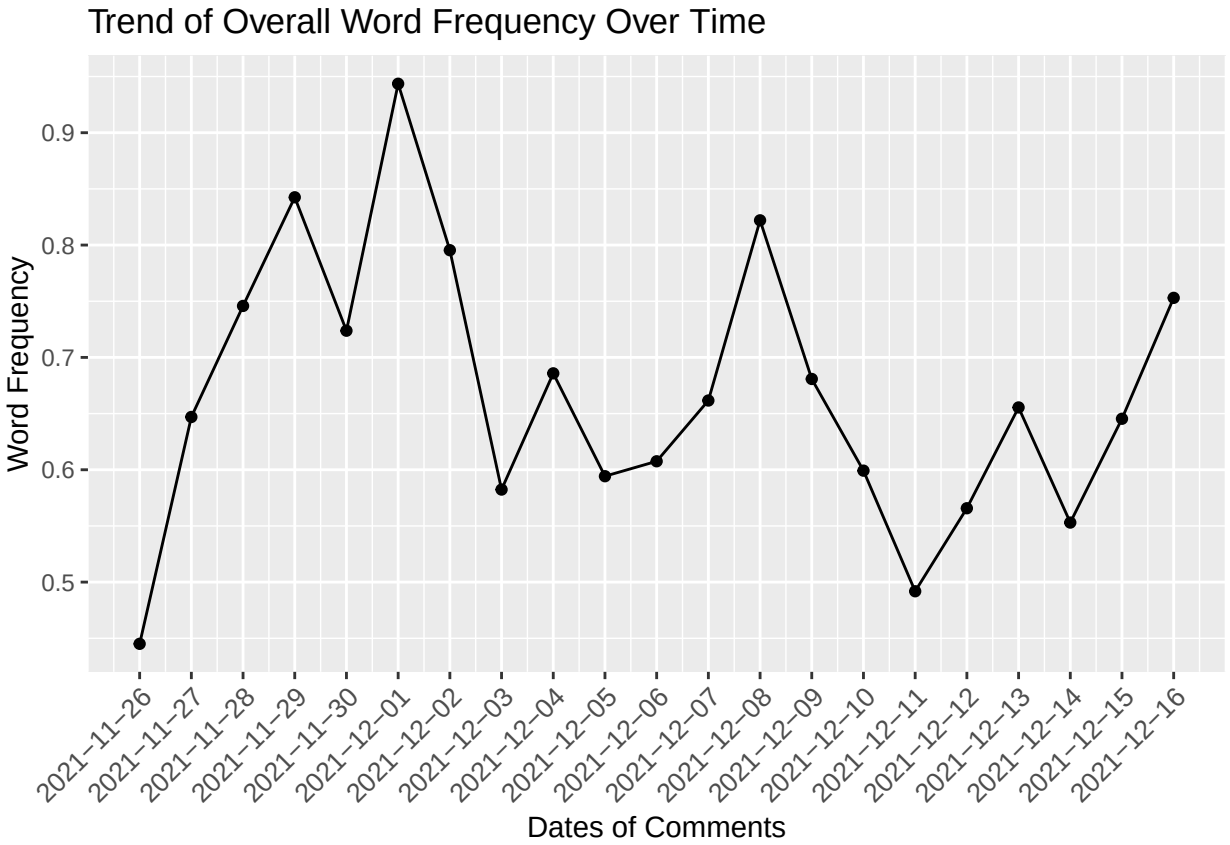
Research question 2: How have the negative moods change in Reddit Comments over time?

We plot the changes from Nov 26 to Dec 16 for word frequency of four words in total and separately.

```
df1 %>%
    filter(word %in% c("bad", "shit", "fuck","die")) %>%
  group_by(date) %>%
  summarise(total = n()) %>%
  ggplot(aes(date,total)) +
  geom_point()+
  geom_line()+
  scale_x_date(breaks = unique(df1$date)) +
  labs(title="Trend of Overall Word Count Over Time",
       x="Dates",
       y="Count")+
  theme(axis.text.x = element_text(angle = 45, size = 10,hjust = 1))
```




```
df4 %>%
  filter(word %in% c("bad", "shit", "fuck","die")) %>%
  group_by(date) %>%
  summarise(total = sum(frequency)) %>%
  ggplot(aes(date,total)) +
  geom_point()+
  geom_line()+
  scale_x_date(breaks = unique(df4$date)) +
  labs(title="Trend of Overall Word Frequency Over Time",
       x="Dates of Comments",
       y="Word Frequency")+
  theme(axis.text.x = element_text(angle = 45, size = 10,hjust = 1))
```



The first plot indicate that overall negative emotion word count (we only include four words mentioned in previous section) peaked during Nov 27 - Nov 29, meaning that a lot of people commented on Omicron-related threads on the Reddit during that period of time. But according to the second plot, although people were more pessimistic compared to several days before, they haven't gone through the toughest time. As we hypothesized, the peak of overall negative words frequency was on Dec 1, when the first Omicron variant cases was found in the US.

We were curious about whether people's negative mood are correlated with new Omicron cases in the US, so we test the relationship between Covid new cases and negative expression. Because we believe Nov 1 is a key date.

```
group1<-df5 %>%
  filter(date<"2021-12-01")
group2<-df5 %>%
  filter(date>="2021-12-01")
summary(lm(total~New.Cases,df5))
```

```
##
## Call:
## lm(formula = total ~ New.Cases, data = df5)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.094936 -0.044387 -0.008735  0.034957  0.171000
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept) 1.511e-01  1.701e-02   8.884 2.75e-13 ***
## New.Cases   1.392e-07  1.430e-07   0.973    0.334
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.05733 on 74 degrees of freedom
## Multiple R-squared:  0.01264,    Adjusted R-squared:  -0.0007062
## F-statistic: 0.9471 on 1 and 74 DF,  p-value: 0.3336
```

```
t.test(group1$total,group2$total)
```

```
##
##  Welch Two Sample t-test
##
## data:  group1$total and group2$total
## t = 0.36842, df = 37.399, p-value = 0.7146
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.02352714  0.03398882
## sample estimates:
## mean of x mean of y
## 0.1702080 0.1649772
```

Both results are not significant, so we don't know whether there are relationships between negative moods and Covid-19 cases.

##Conclusion and Limitation

As for the first research question, we found that word "bad" and "shit" are always more popular than other two words, maybe indicating the anger emotion. However, if we shift our gaze to a broad picture, a different answer can be presented. Fear becomes the dominant mood for reddit users, followed by sadness, negativeness, and disgust. Regarding the second research question,we provide data visualizations implying the change of number and frequency of negative moods words occurred, as well as showing the highest frequency of words within different dates. Unfortunately,our result shows that there's no significance for the relationship between these characteristics. This is possibly a consequence of some limitations of our data processing process. Firstly, We are unable to separate the Omicron data from the overall Covid-19 epidemic data for independent analysis. Without isolating the cases of the new variant, it is unrealistic to accurately compare the causes of people's negative moods during different periods of time. Secondly, we only use linear regression model to analyze the correlation. A more advanced model is suggested to use for figuring out the relationship between different periods and # and freq of negative moods words.