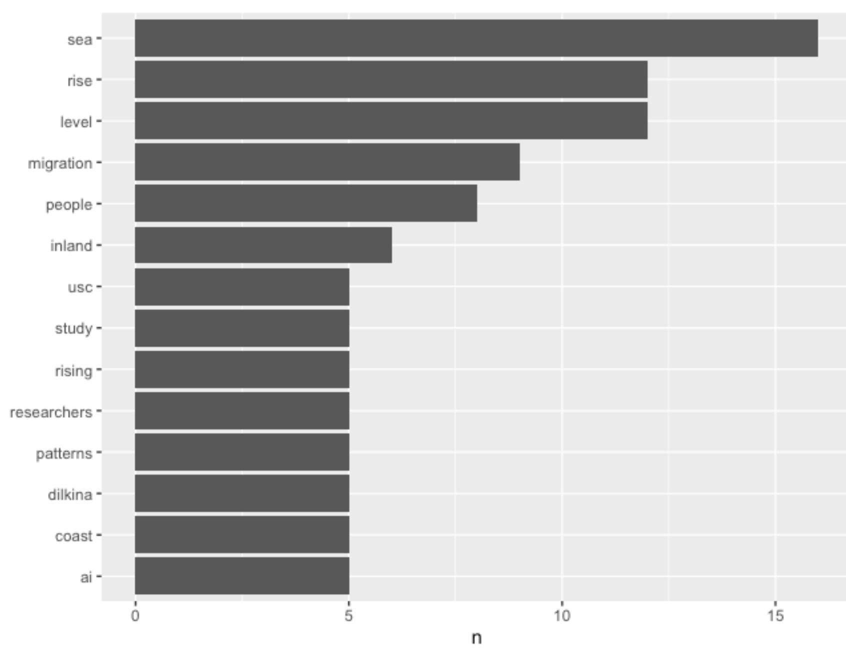
Sally Moywaywa
Business Insight Report

An analysis into sea level rising in the United States

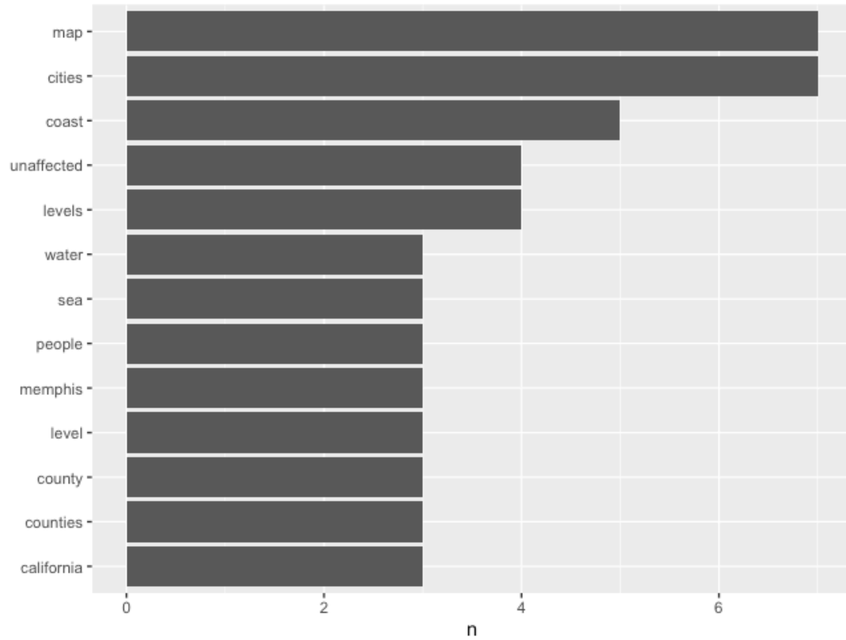
"Climate change is no longer some far-off problem; it is happening here; it is happening now." (Barack Obama- Former President of the United States of America on Climate change)
It is no longer a secret that climate change is real, the negative effects of climate change are being felt all over the world through different phenomenon like increased forest fires, severe drought, devastating flooding and most importantly through the rising sea level. Countries like Tuvalu, Kiribati, Marshall Islands and Indonesia battling to stay afloat, and it is not just these pacific islands that are rapidly sinking but also major cities in the United states such as Miami, Houston and New Orleans. As these cities in America sink faster than ever at an approximated rate of 2 inches per year, they are becoming more and more vulnerable to increasing disasters such as hurricanes. It is predicted that by 2100, millions of people will be forced to relocate to non-coastal cities due to rising sea levels. In this report I analyze how American citizens feel in regard to having to relocate from some of the most populated coastal cities and what sentiments they have towards climate change.

First, I got an article about how climate change-driven sea level rise could trigger relocation in the United States and comments from reddit on the specific article to analyze sentiments.
I then read the two datasets, tokenized them, removed the stop words, got the most frequent tokens from each then plotted a token frequency histogram for each of the data sets.



(article token frequency histogram)

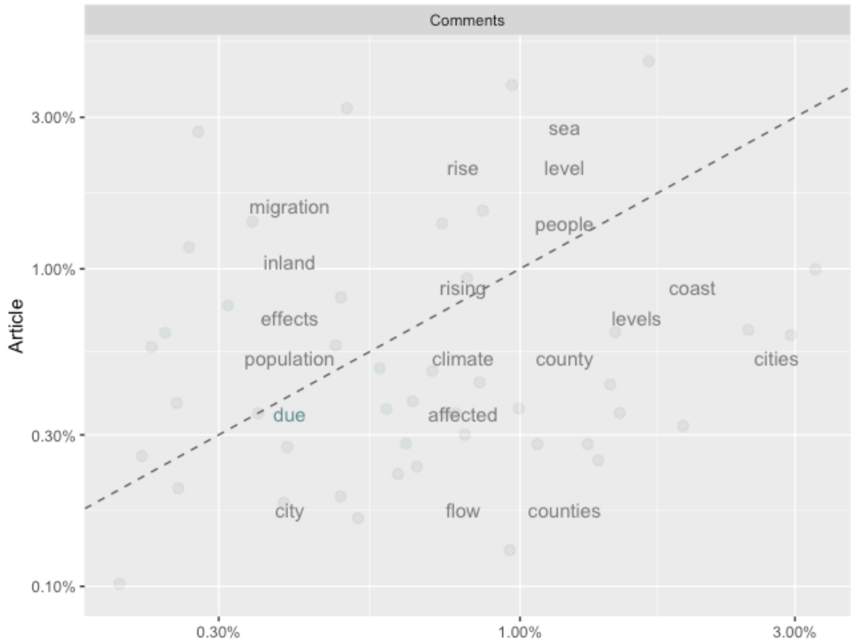
From the article token frequency histogram, it is evident that sea, level, rise, migration, people and inland are the most frequent token used. This suggests that the article talks about the inland migration of people due to sea level rise in the United States.



(comments token frequency histogram)

From the comments token frequency histogram however, map, cities, coast and unaffected are the most frequent tokens. This suggests that most reddit users on reading the article immediately sought to see which cities were affected (mostly coastal cities were affected). This could probably be because human beings tend to care to see if they and their loved ones would be affected by this shift.

I then combined the two datasets, plotted a correlogram and did a correlation test to see how similar the two datasets were.
From the correlogram, it was evident that the two datasets talked of similar topics since the words near the line like rising, population, people are found in both datasets. However, keen differences between the article and reddit reactions to the article are again seen. While the article focuses majorly on the fact that the sea level is rising, and inland migration is bound to happen. The reddit comments to the article focus on majorly the affected cities, counties which are primarily in the coastal regions in the United States. This is why when a correlation test is done, the correlation coefficient is 0.2462 showing a weak but positive correlation between the two articles.



(article- comments correlogram)

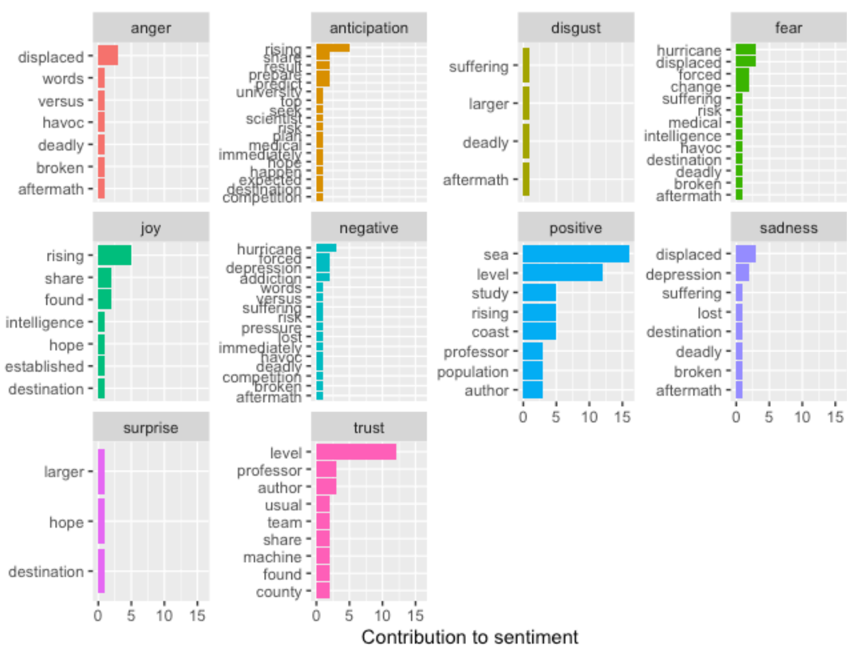
I then did a sentiment analysis on the article to see what emotions the article provoked. From the NRC sentiment word cloud, it can be seen that most of the words in the article are taken to be positive, evoking emotions of trust with a little anticipation.

(article's NRC sentiment word cloud)

However, using the Bing sentiment lexicon, the article is seen to be more negative.

(article's Bing sentiment word cloud)

A further analysis on the article's NRC sentiment lexicon using a graph and grouping the words by sentiment showed that most words were negative and evoked fear and anticipation. However a closer look would show you that this results cannot be extremely reliable because words such as sea, level, rising, coast are seen to be positive but we know that we are talking about the sea level rising and when these words are used together they may evoke a different feeling. Hence, I did a further analysis using a bigram and not just single tokens.



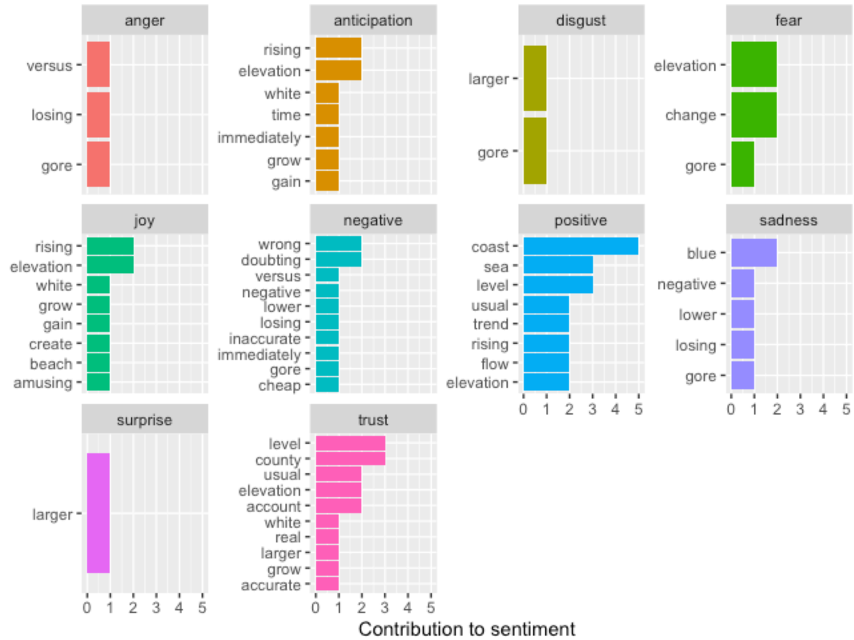
(article's NRC sentiment analysis- and their contribution to each sentiment)

Analysis of the bigram sentiments using the NRC lexicon showed that the article provoked emotions of fear, anticipation, anger when referring to the rising sea level, and to climate change and thoughts of displacement and relocation.

I then did a sentiment analysis on the reddit comments. An analysis using the NRC sentiment lexicon showed almost similar results as the articles with most of the words being positive, evoking emotions of trust and some anticipation.

(comment's NRC sentiment word cloud)

However, when grouping the comment tokens using the NRC lexicon by sentiment, most of the words are seen to be negative. We still have a similar problem of vagueness when using one token in the analysis. This necessitated the use of a bigram in the sentiment analysis.
From the bigram, we are able to see that rising sea levels are associated with negative sentiments and evoke emotions of fear and anticipation. We are also able to see names of counties such as Jefferson county and Shelby county which are both in Texas and were hit hard by hurricane Harvey in 2017, therefore showing the negative impact of climate change and sea level rise in the United States.



(comment's NRC sentiment analysis- and their contribution to each sentiment)

From the analysis above it can be seen that climate change- driven sea level rise triggers emotions of fear, anticipation, sadness and anger, all which are negative among Americans. However, the fact that studies and research are being done to inform, educate and look for solutions to slow down and eventually dampen climate change makes the more positive, joyous and trustful in researchers and the government for they hope that a solution will be found in the near future.

Reference:

Dawson, C. (n.d.). Sea-Level Rise Could Reshape the United States, Trigger Migration Inland - USC Viterbi: School of Engineering. Retrieved from https://viterbischool.usc.edu/news/2020/01/sea-level-rise-could-reshape-the-united-states-trigger-migration-inland/

Climate change-driven sea-level rise could trigger mass migration of Americans to inland cities. A new study uses machine learning to project migration patterns resulting from sea-level rise. (2020, January 25). Retrieved February 14, 2020, from https://www.reddit.com/r/science/comments/etue8q/climate_changedriven_sealevel_rise_could_trigger/

Worland, J. (2019, June 13). How Leaders of Sinking Countries Are Fighting Climate Change. Retrieved from https://time.com/longform/sinking-islands-climate-change/

Lakritz, T. (2019, November 13). 11 sinking cities that could soon be underwater. Retrieved February 14, 2020, from https://www.insider.com/sinking-cities-climate-change-2019-5#virginia-beach-virginia-6

<u>**APPENDIX**</u>

R code:
```
#loading packages
library(rvest)
library(tidyverse)
library(tidytext)
library(dplyr)
library(tidyr)
library(scales)
library(ggplot2)
library(textdata)
library(stringr)
library(wordcloud)
library(reshape2)

#reading article comments
article <- read_html("https://viterbischool.usc.edu/news/2020/01/sea-level-rise-could-reshape-
the-united-states-trigger-migration-inland/")

#getting article title
articletitle <- article%>%
  html_nodes("title")%>%
  html_text()

#viewing article title
articletitle
```

R Output:
```
articletitle
[1] "Sea-Level Rise Could Reshape the United States, Trigger Migration Inland - USC Viterbi |
School of Engineering"
```

R code:
```
#getting text from the article webpage
articletxt <- article%>%
  html_nodes("p")%>%
  html_text()

#reading opinions about the article from reddit webpage
climate_change <-
read_html("https://www.reddit.com/r/science/comments/etue8q/climate_changedriven_sealevel_
rise_could_trigger/")

#getting the title of the discussion
climate_change%>%
```

```
  html_node("title")%>%
  html_text()
```

R Output:

[1] "Climate change-driven sea-level rise could trigger mass migration of Americans to inland cities. A new study uses machine learning to project migration patterns resulting from sea-level rise. : science"

R code:

```
#getting comments from the discussion
comments_data <-climate_change%>%
  html_nodes("p._1qeIAgB0cPwnLhDF9XSiJM")%>%
  html_text()

#changing the data to a vector
comments_data1 <- c(comments_data)

#making a vector from the article
articledf <- data.frame(line= 1, text= articletxt, stringsAsFactors = FALSE)

#making a data frame from comments
commentdf <-  data.frame(id = 1:19, text = comments_data1, stringsAsFactors = FALSE)

#tokenizing comments dataframe
comments_token <-commentdf%>%
  unnest_tokens(word, text)%>%
  anti_join(stop_words)%>%
  count(word,sort =TRUE)

#printing frequent comment tokens
comments_token
```

R Output:

```
# A tibble: 181 x 2
   word          n
   <chr>      <int>
 1 cities        7
 2 map           7
 3 coast         5
 4 levels        4
 5 unaffected    4
 6 california    3
 7 counties      3
 8 county        3
```

```
 9 level          3
10 memphis        3
# … with 171 more rows
```

R code:

```
#tokenizing article dataframe
article_token <- articledf%>%
  unnest_tokens(word, text)%>%
  anti_join(stop_words)%>%
  count(word, sort = TRUE)

#printing frequent comment tokens
article_token
```

R Output:

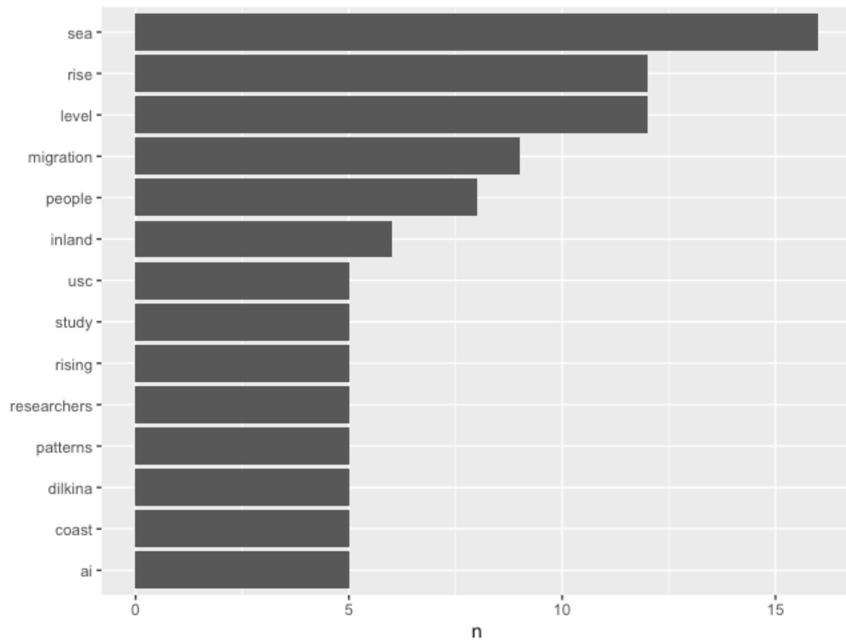
```
# A tibble: 332 x 2
   word         n
   <chr>     <int>
 1 sea          16
 2 level        12
 3 rise         12
 4 migration     9
 5 people        8
 6 inland        6
 7 ai            5
 8 coast         5
 9 dilkina       5
10 patterns      5
# … with 322 more rows
```

R code:

```
#plotting token frequency histograms

#articles token frequency histogram
article_hist <- articledf%>%
  unnest_tokens(word, text)%>%
  anti_join(stop_words)%>%
  count(word, sort=TRUE) %>%
  filter(n > 4) %>% # we need this to eliminate all the low count words
  mutate(word = reorder(word,n )) %>%
  ggplot(aes(word, n))+
  geom_col()+
  xlab(NULL)+
  coord_flip()

#printing the comment frequency token histogram
```

print(article_hist)

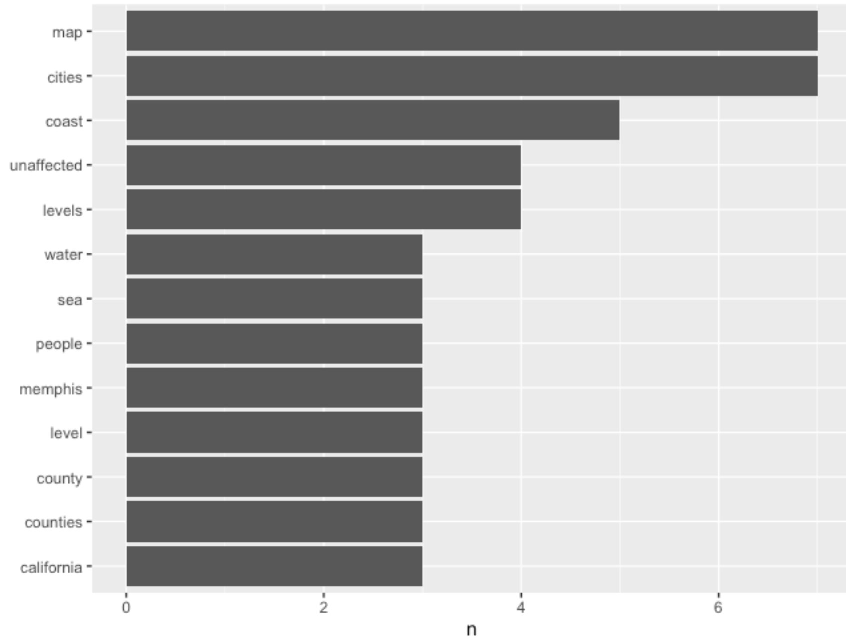
R Output:



R code:

```
#comments token frequency histogram
comment_hist <- commentdf%>%
  unnest_tokens(word, text)%>%
  anti_join(stop_words)%>%
  count(word, sort=TRUE) %>%
  filter(n > 2) %>% # we need this to eliminate all the low count words
  mutate(word = reorder(word,n )) %>%
  ggplot(aes(word, n))+
  geom_col()+
  xlab(NULL)+
  coord_flip()

#printing the comment frequency token histogram
comment_hist
```

R Output:



R code:

```
#removing stop words
tidyarticle<- articledf%>%
  unnest_tokens(word, text)%>%
  anti_join(stop_words)

tidycomments <-commentdf%>%
  unnest_tokens(word, text)%>%
  anti_join(stop_words)


#combining the two data sets
combo_data <- bind_rows(mutate(tidycomments, author="Comments"),
                        mutate(tidyarticle, author= "Article")
)%>%#closing bind_rows
  mutate(word=str_extract(word, "[a-z']+")) %>%
  count(author, word) %>%
  group_by(author) %>%
  mutate(proportion = n/sum(n))%>%
  select(-n) %>%
  spread(author, proportion) %>%
  gather(author, proportion, `Comments`)

combo_data
```

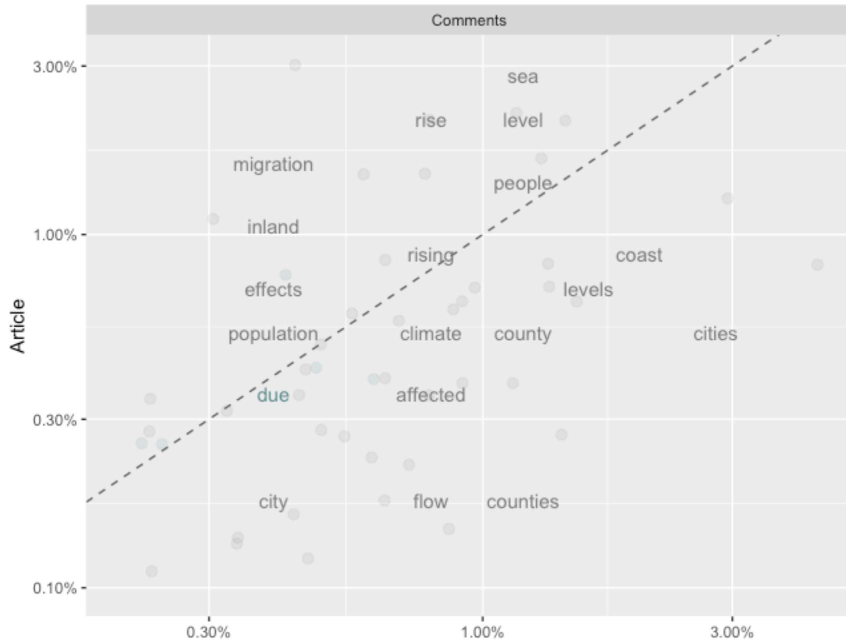
R Output:

```
# A tibble: 452 x 4
   word       Article author   proportion
   <chr>        <dbl> <chr>        <dbl>
 1 abilities  0.00193 Comments   NA
 2 accept     0.00193 Comments   NA
 3 account   NA       Comments    0.00797
 4 accuracy  NA       Comments    0.00398
 5 accurate  NA       Comments    0.00398
 6 actively  NA       Comments    0.00398
 7 activity   0.00193 Comments   NA
 8 adapt     NA       Comments    0.00398
 9 added      0.00193 Comments   NA
10 addiction  0.00387 Comments   NA
# … with 442 more rows
```

R code:

```
#plotting the correlograms
ggplot(combo_data, aes(x=proportion, y=`Article`,
                      color = abs(`Article`- proportion)))+
  geom_abline(color="grey40", lty=2)+
  geom_jitter(alpha=.1, size=2.5, width=0.3, height=0.3)+
  geom_text(aes(label=word), check_overlap = TRUE, vjust=1.5) +
  scale_x_log10(labels = percent_format())+
  scale_y_log10(labels= percent_format())+
  scale_color_gradient(limits = c(0,0.001), low = "darkslategray4", high = "gray75")+
  facet_wrap(~author, ncol=2)+
  theme(legend.position = "none")+
  labs(y= "Article", x=NULL)
```

R Output:



R code:

cor.test(data=combo_data[combo_data$author == "Comments",],
~proportion + `Article`)

R Output:

Pearson's product-moment correlation

data: proportion and Article
t = 1.6854, df = 44, p-value = 0.09899
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
-0.0474255 0.5007617
sample estimates:
cor
0.246263

R code:

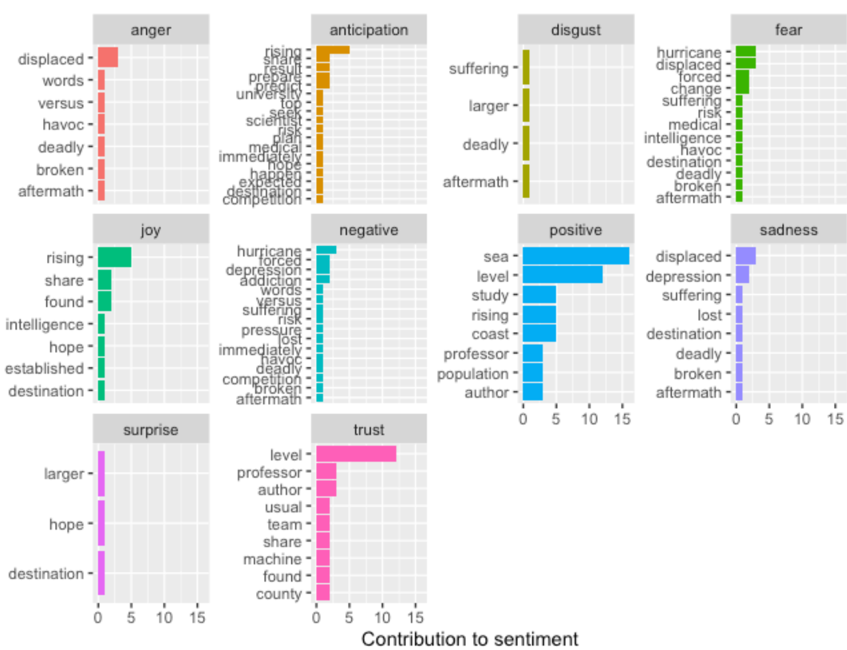
#SENTIMENT ANALYSIS

#getting the sentiments
afinn<- get_sentiments("afinn")
nrc<- get_sentiments("nrc")
bing<- get_sentiments("bing")

#####ARTICLE

```
tidyarticle %>%
  inner_join(get_sentiments("afinn"))%>%
  summarise(sentiment=sum(value)) %>%
  mutate(method="AFINN")

tidyarticle %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort=T)

articlesentiment <-tidyarticle %>%
  inner_join(get_sentiments("nrc")) %>%
  count(word, sentiment, sort=T) %>%
  ungroup()

articlesentiment  %>%
  group_by(sentiment) %>%
  top_n(8) %>%
  ungroup() %>%
  mutate(word=reorder(word, n)) %>%
  ggplot(aes(word, n, fill=sentiment)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~sentiment, scales = "free_y")+
  labs(y="Contribution to sentiment", x=NULL)+
  coord_flip()
```

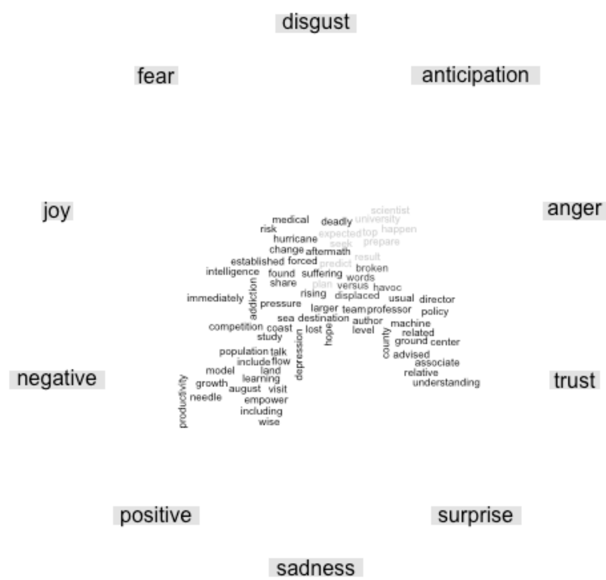
R Output:



R code:

```
##### article word cloud

tidyarticle %>%
  inner_join(get_sentiments("nrc")) %>%
  count(word, sentiment, sort=TRUE) %>%
  acast(word ~sentiment, value.var="n", fill=0) %>%
  comparison.cloud(colors = c("grey20", "gray80"),
                   max.words=100,
                   scale = c(0.5,0.5),
                   fixed.asp = TRUE,
                   title.size = 1)
```

R Output:



R code:

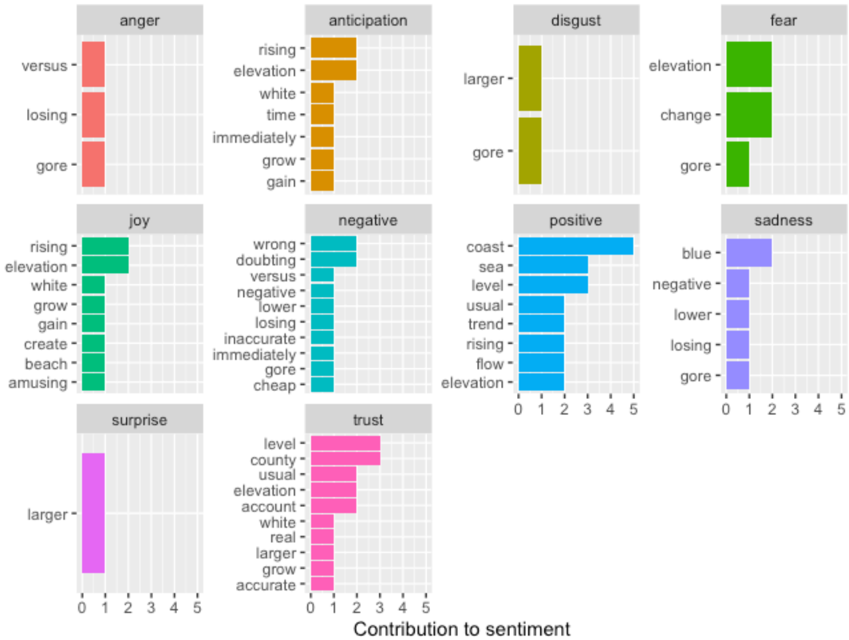
```
tidyarticle %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort=TRUE) %>%
  acast(word ~sentiment, value.var="n", fill=0) %>%
  comparison.cloud(colors = c("grey20", "gray80"),
                   max.words=100,
                   scale = c(1.0,1.0),
                   fixed.asp = TRUE,
                   title.size = 1)
```

R Output:

R code:

```
###sentiment analysis of the comments

tidycomments %>%
  inner_join(get_sentiments("afinn"))%>%
  summarise(sentiment=sum(value)) %>%
  mutate(method="AFINN")

tidycomments %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort=T)

commentsentiment <-tidycomments%>%
  inner_join(get_sentiments("nrc")) %>%
  count(word, sentiment, sort=T) %>%
  ungroup()

commentsentiment  %>%
  group_by(sentiment) %>%
  top_n(6) %>%
  ungroup() %>%
  mutate(word=reorder(word, n)) %>%
  ggplot(aes(word, n, fill=sentiment)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~sentiment, scales = "free_y")+
  labs(y="Contribution to sentiment", x=NULL)+
  coord_flip()
```

R Output:



R code:

```
##### comments word cloud

tidycomments %>%
  inner_join(get_sentiments("nrc")) %>%
  count(word, sentiment, sort=TRUE) %>%
  acast(word ~sentiment, value.var="n", fill=0) %>%
  comparison.cloud(colors = c("grey20", "gray80"),
                   max.words=100,
                   scale = c(0.5,0.5),
                   fixed.asp = TRUE,
                   title.size = 1)
```

R Output:

R code:

```
tidycomments %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort=TRUE) %>%
  acast(word ~sentiment, value.var="n", fill=0) %>%
  comparison.cloud(colors = c("grey20", "gray80"),
                   max.words=100,
                   scale = c(1.0,1.0),
                   fixed.asp = TRUE,
                   title.size = 1)
```

R Output:

R code:

```
####Bigrams

###Article
Article_bigram <- articledf %>%
  unnest_tokens(bigram, text, token = "ngrams", n=2) %>%
  separate(bigram, c("word1", "word2"), sep = " ") %>%
  filter(!word1 %in% stop_words$word) %>%
  filter(!word2 %in% stop_words$word) %>%
  count(word1, word2, sort = TRUE)

Article_bigram

# sentiment analysis on the article bigram
articlesentiment2 <-Article_bigram %>%
  inner_join(get_sentiments("nrc"), by = c(word2 = "word")) %>%
  count(word1, word2, sentiment, sort=T) %>%
  ungroup()

articlesentiment2
```

R Output:

```
# A tibble: 83 x 4
   word1      word2       sentiment     n
   <chr>      <chr>       <chr>     <int>
 1 2017       displaced   anger         1
 2 2017       displaced   fear          1
 3 2017       displaced   sadness       1
 4 ai         study       positive      1
 5 artificial intelligence fear         1
 6 artificial intelligence joy          1
 7 artificial intelligence positive     1
 8 artificial intelligence trust        1
 9 assistant  professor   positive      1
10 assistant  professor   trust         1
# … with 73 more rows
```

R code:

```
### comments
Comments_bigram <- commentdf %>%
  unnest_tokens(bigram, text, token = "ngrams", n=2) %>%
  separate(bigram, c("word1", "word2"), sep = " ") %>%
  filter(!word1 %in% stop_words$word) %>%
  filter(!word2 %in% stop_words$word) %>%
  count(word1, word2, sort = TRUE)
```

```
Comments_bigram

# sentiment analysis on the article bigram
commentsentiment2 <-Comments_bigram %>%
  inner_join(get_sentiments("nrc"), by = c(word2 = "word")) %>%
  count(word1, word2, sentiment, sort=T) %>%
  ungroup()

commentsentiment2
```

R Output:

```
# A tibble: 21 x 4
   word1      word2  sentiment     n
   <chr>      <chr>  <chr>     <int>
 1 actively   losing anger         1
 2 actively   losing negative      1
 3 actively   losing sadness       1
 4 al         gore   anger         1
 5 al         gore   disgust       1
 6 al         gore   fear          1
 7 al         gore   negative      1
 8 al         gore   sadness       1
 9 california coast  positive      1
10 climate    change fear          1
# … with 11 more rows
```