Textmining

Sky Liu

November 4, 2018

SCRAPING DATA FROM https://correlaid.org/blog/

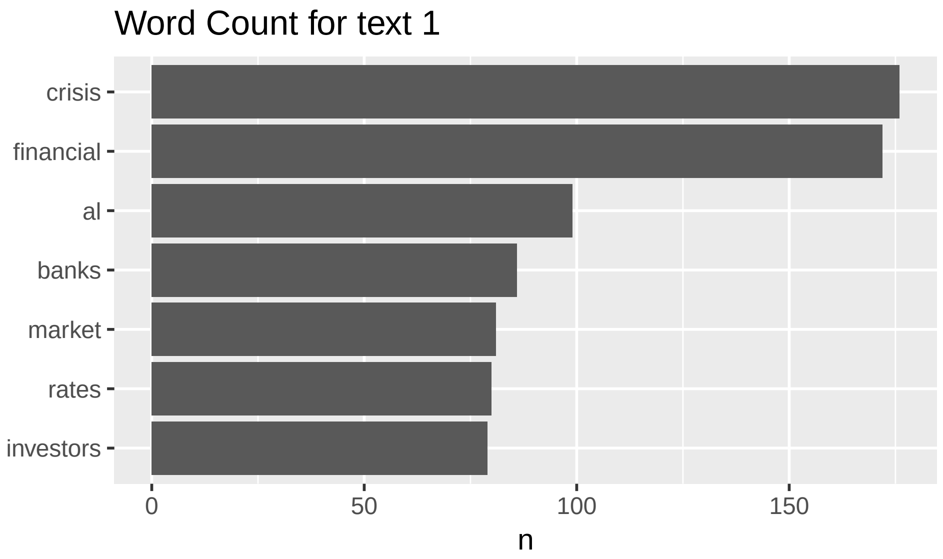
```r
#get text from the blog
#read text
text1 <- read.table(file ="NAEC_Origins-of-the-Crisis_ENG.txt",header = FALSE,sep="\n")
```

GET A TIDY TEXT FORMAT & WORD COUNT

```r
#remove empty lines
text1 <- text1 %>% filter(V1 != " ")
text1 <- text1 %>% filter(V1 != "\f")
colnames(text1) <- "text"
line <- c(1:length(text1))
text1 <- cbind(line,text1)
text1$text<-as.character(text1$text)
#a token per row
text1 <-text1 %>%unnest_tokens(word,text)
#get rid of any non-characters
text1 <- text1 %>%mutate(word = str_extract(word,"[a-z']+"))
text1 <-na.omit(text1)
#get rid of stop-words
text1<- text1 %>% anti_join(stop_words)
```

```
## Joining, by = "word"
```

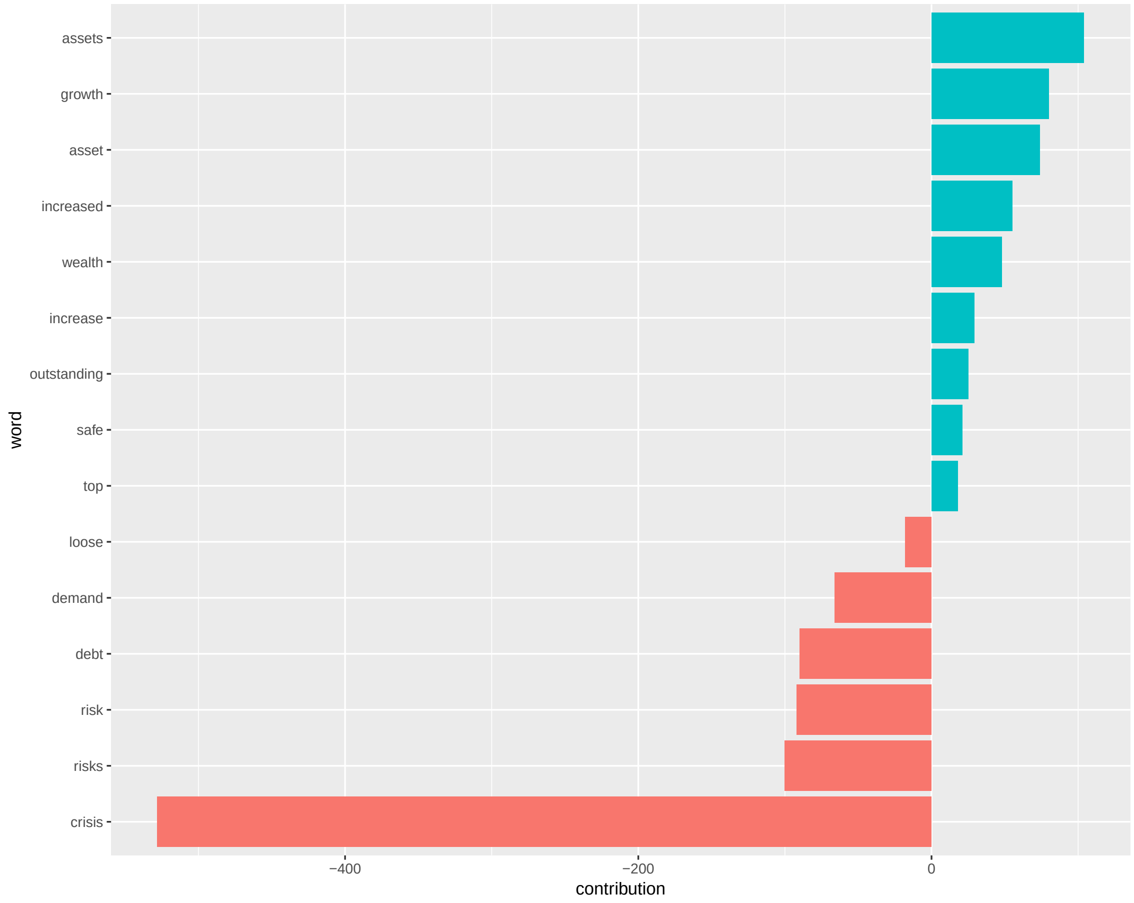
```r
s<-stop_words
#word count
text1 %>%
count(word, sort = TRUE)%>%
  filter(n > 75) %>%
  mutate(word = reorder(word, n))%>%
  ggplot(aes(word, n)) +geom_col() +xlab(NULL) +coord_flip() +ggtitle("Word Count for text 1")
```



Sentiment Analysis With Tidy Data

```
contributions <- text1 %>%
  inner_join(get_sentiments("afinn"), by = "word") %>%
  group_by(word) %>%
  summarize(occurences = n(),
            contribution = sum(score))

contributions %>%
  top_n(15, abs(contribution)) %>%
  mutate(word = reorder(word, contribution)) %>%
  ggplot(aes(word, contribution, fill = contribution > 0)) +
  geom_col(show.legend = FALSE) +
  coord_flip()
```



Comparing the three sentiment dictionaries

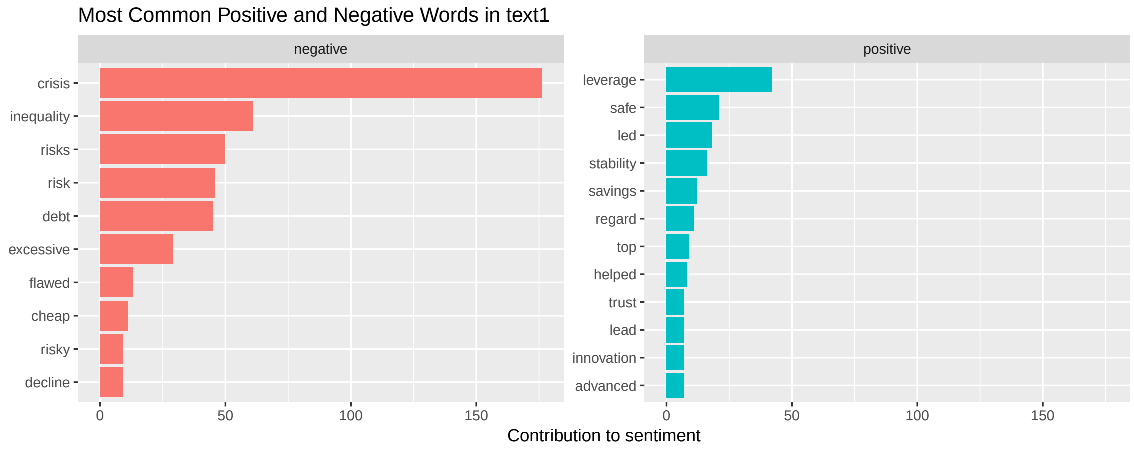
Most common positive and negative words

```
bing_word_counts <- text1 %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = TRUE) %>%
  ungroup()
```

```
## Joining, by = "word"
```

```
bing_word_counts %>%
  group_by(sentiment) %>%
  top_n(10) %>%
  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() + ggtitle("Most Common Positive and Negative Words in text1")
```

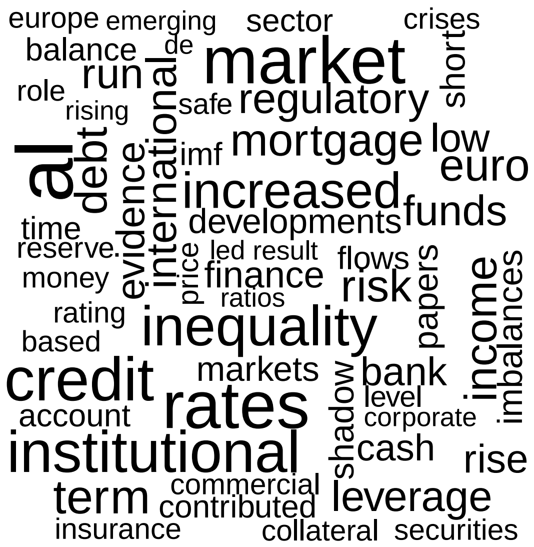
```
## Selecting by n
```



Word Cloud

```
text1 %>%
  anti_join(stop_words) %>%
  count(word) %>%
  with(wordcloud(word, n, max.words = 100))
```

```
## Joining, by = "word"
```



Bigram sentiment Analysis

```
#taking bigram and filtering out non character and OAs
text11<-read.table(file ="NAEC_Origins-of-the-Crisis_ENG.txt",header = FALSE,sep="\n")

text11 <- text11 %>% filter(V1 != " ")
text11 <- text11 %>% filter(V1 != "\f")
colnames(text11) <- "text"
```

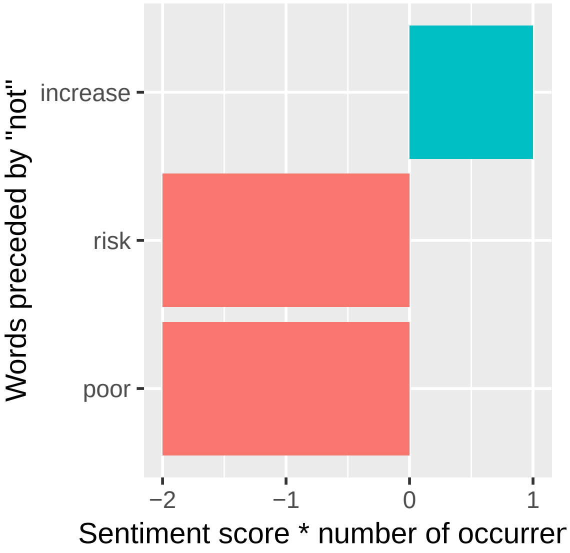
```r
line <- c(1:length(text11))
text11 <- cbind(line,text11)
text11$text<-as.character(text11$text)

text11<- text11%>%unnest_tokens(bigram, text, token = "ngrams", n = 2)%>%
  separate(bigram, c("word1", "word2"), sep = " ") %>%
  mutate(word1 = str_extract(word1,"[a-z']+"))%>%
  na.omit(word1)%>%
  mutate(word2 = str_extract(word2,"[a-z']+"))%>%
  na.omit(word2)

negation_words <- c("not", "no", "never", "without","don't")

#filter out bigrams starts with negation words
negation_words <- text11 %>%
  filter(word1 %in% negation_words) %>%
  inner_join(get_sentiments("afinn"), by = c(word2 = "word")) %>%
  count(word1, word2, score, sort = TRUE) %>%
  ungroup()


#plot negation words
 negation_words %>%
  mutate(contribution = n * score) %>%
  arrange(desc(abs(contribution))) %>%
  head(20) %>%
  mutate(word2 = reorder(word2, contribution)) %>%
  ggplot(aes(word2, n * score, fill = n * score > 0)) +
  geom_col(show.legend = FALSE) +
  xlab("Words preceded by \"not\"") +
  ylab("Sentiment score * number of occurrences") +
  coord_flip()
```


Words preceded by "not"
increase
risk
poor
Sentiment score * number of occurrer