Text Analytics
&
Natural Language Processing

Assignment

Business Insight Report

Instructor

Prof. Thomas Kurnicki

Cohort

FMSBA2

Student

Baturalp Topcu

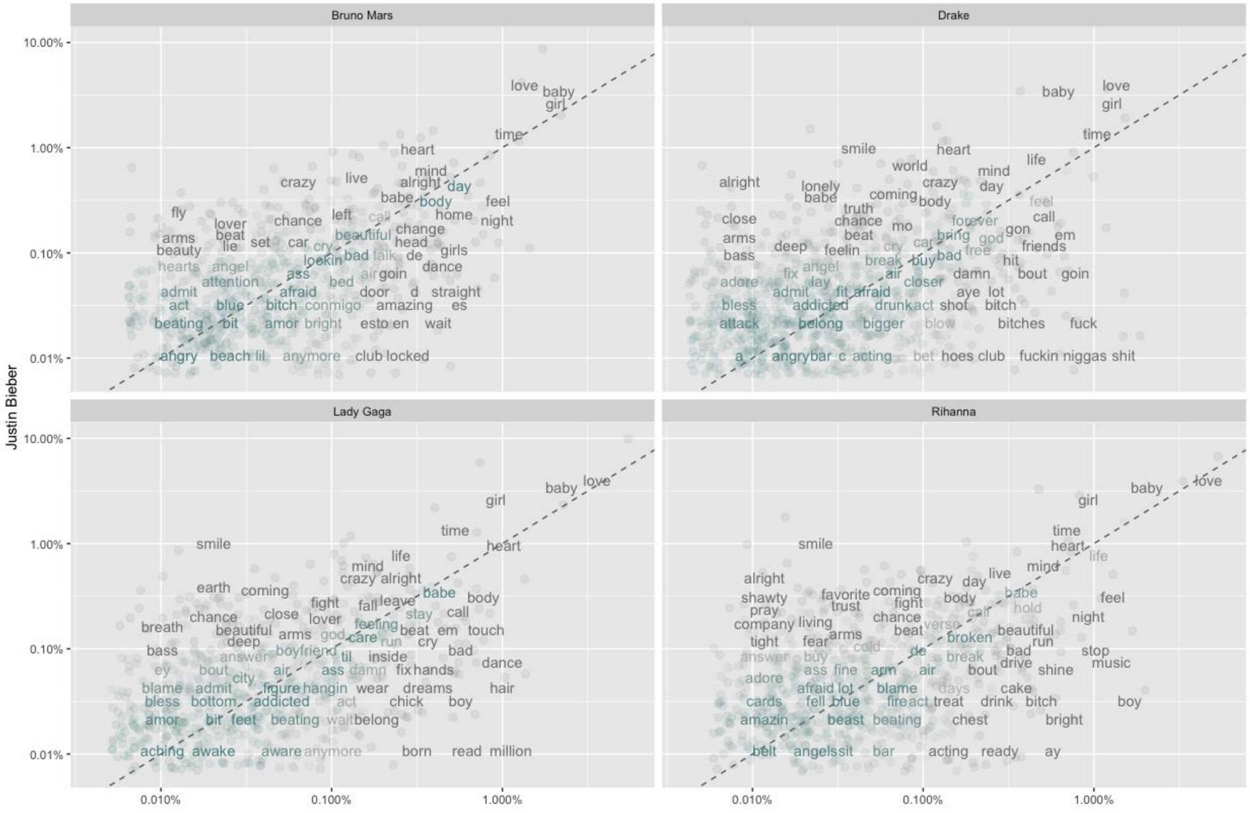
4905507

Text Analytics Report of the Top 5 Musician in the Last Decade

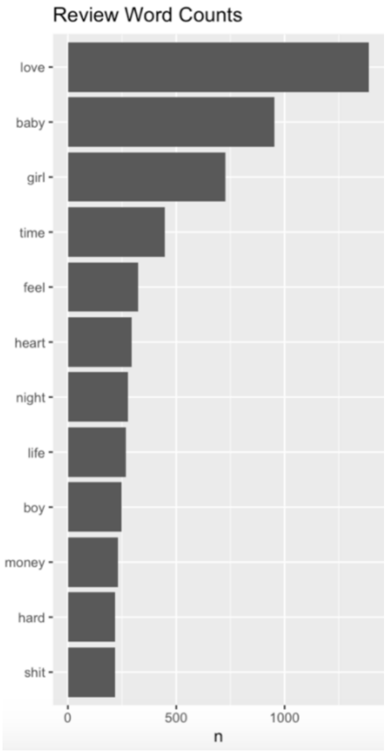
My goal to prepare this report is to see being successful in music industry depends on luck or on a secret formula. To reach a reliable result, I choose top 5 musician for the last decade; Justin Bieber, Drake, Bruno Mars, Lady Gaga and Rihanna. I created a data set by collecting the lyrics of these five musicians to see what makes them successful.

To learn most up to date number of listeners, I took monthly listeners number from Spotify and number of followers on Instagram. Drake has 52 million, Bruno Mars has 32 million, Lady Gaga has 29 million, Rihanna has 33 million and Justin Bieber has 62 million monthly listeners on Spotify. On Instagram, Drake has 63 million, Bruno Mars has 22 million, Lady Gaga has 39 million, Rihanna has 79 million and Justin Bieber has 127 million followers. These numbers clearly show that Justin Bieber is the most successful artist in the music industry by a long way. After him, Drake and Rihanna take place. To understand the facts, I started my analysis by comparing their lyrics to each other to see if they use similar wording. All of the five musicians use almost same wording, their songs are mostly about love, daily life, sexual life, feelings, money and friends. These 5 musicians are highly correlated to each other. Justin Bieber

Correlograms of Top Five Artist in the Last Decade



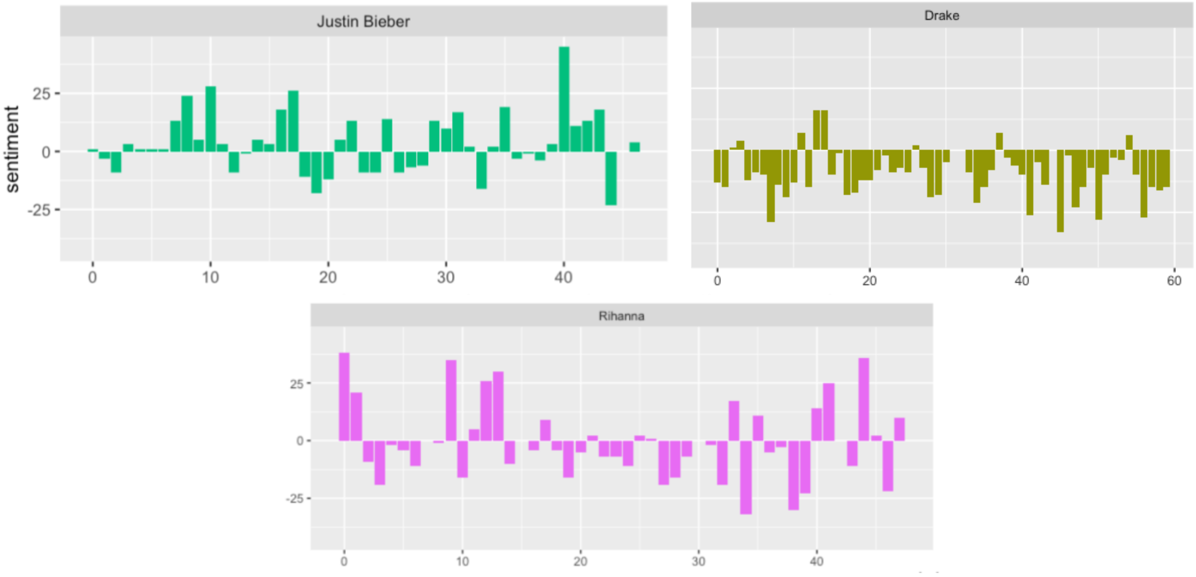
I went deeper to understand is there any formula or way to write a lyric which can carry an unknown artist to top ten list. I analyzed the most used words in all of the five musicians' lyrics. 'Love' and 'baby' are the most used word in popular songs but it's too general to understand details of the successful lyric's formula. The frequency of 'boy' and 'girl' catch my attention because, while 'girl' is used by all of them, 'boy' is just used by Lady Gaga and Rihanna. So, it may be the sign of a sexual preference or a way of storytelling. When we combine other words; 'boy', 'girl', 'baby', 'night', 'life' and 'heart', they create an ambience of a person who goes out to enjoys the night life and looks for a partner to fall in love.



The last three words in the chart points out the hardship and the reality of the life because everyone's live has ups and downs. Writing sincere lyric help musicians to develop empathy with their listeners which effect their career positively.

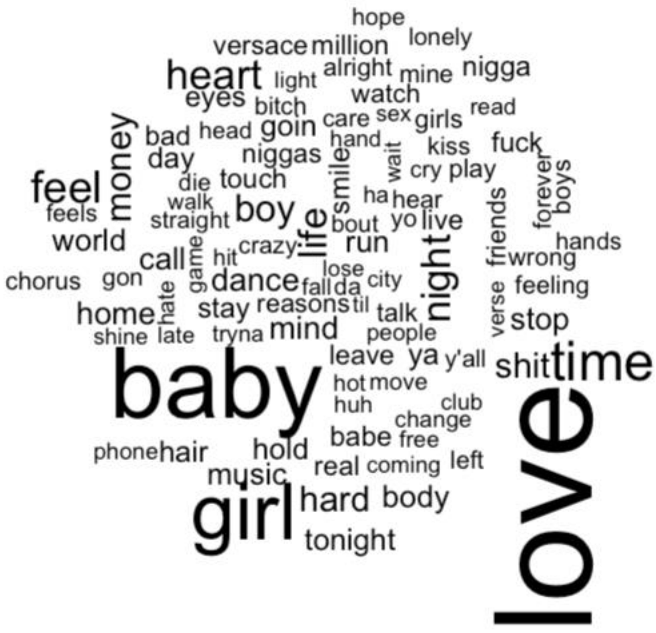
According to Dr.Chappel, music makes you happier, reduces stress and improves health because music decreases the level of stress hormone in our body and help to overcome depression(Chappel,2019). Bethancourt explains how songs make happier and reduce stress wonderfully. Sometimes the song that you listen, or sing has the sentences or feelings you can't share with anyone. One lyric can be your and others voice. In this way, you cannot feel alone and depress because it's the way that you share your love, life and sometimes disappointments (Bethancourt,2017).

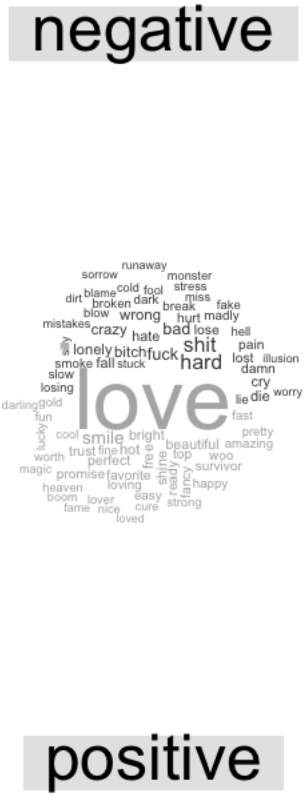
Negative - Positive Sentiment Analysis (Bing)



Based on the data from Spotify and Instagram, Justin Bieber, Drake and Rihanna are the most popular artist. To understand what kind of feelings, attract listeners, I conducted a ‘bing’ sentiment analysis above. My observation from the sentiment analysis, Justin Bieber uses mostly positive words to create positive mood on his listeners. Basically, I can say that people listen Justin Bieber’s songs to boost their mood or because they are happy and shares the same emotion. On the other hand, Drake’s songs have more negative words and feelings. I can state that the fan of Drake are the people who are more under stress and they have more negative feelings. Therefore, they listen Drake to reduce their stress by empathizing with his songs. Rihanna’s chart is a balanced regarding to negativity and positivity. To understand her fans’ segment, there should be further analysis based on years, trends, and her life. However, this chart is explaining why the number of followers on Instagram is between Justin Bieber and Drake, while the monthly listeners’ number lower than both.

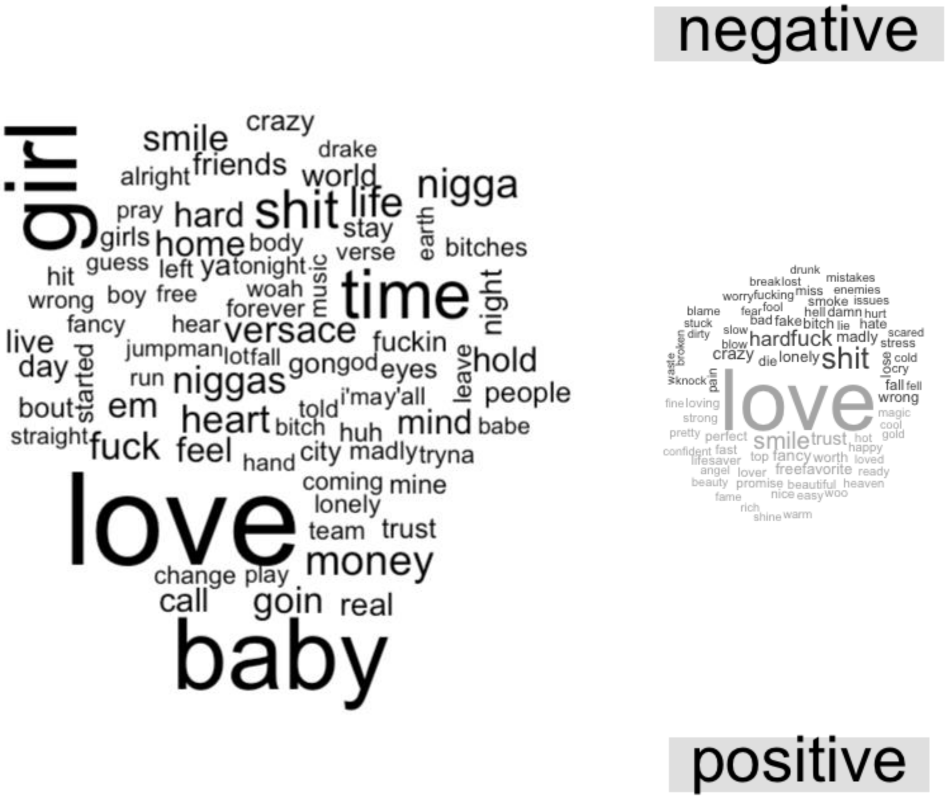
The Top Five Musicians’ Word Cloud





I created two separated word clouds, the one above shows the words used by all five musicians and the one below is from the two completely opposite musician in terms of positive and negative lyrics. The most popular five musician mostly focus on love, sexual life, emotion, gender identity, aggressivity and loneliness. When we look at the Drake and Justin Bieber word cloud, they have the same themes as others but now we can see that new themes like friendship, trust, money and swearing. It explains that, since your lyrics are sincere and if the words give people what they want to feel or share to be happier or to get relax, this is a potential promising lyric to carry a musician to top five list. As a result of our analysis, I suggest musicians or professions in music industry to focus on either positive feelings or negative feelings because the sentiment analysis shows that the musician who try to balance emotions lack behind the musician who produce just positive or just negative mood songs.

Drake and Justin Bieber Word Cloud



Reference

Chappel, M., M,.(2019) “Scientists Find 15 Amazing Benefits of Listening to Music.” Retrieved from https://www.lifehack.org/317747/scientists-find-15-amazing-benefits-listening-music

Bethancourt,A.,B.,(2017) “Why the Lyrics are Important.” Retrieved from https://www.theodysseyonline.com/why-the-lyrics-are-important

Appendix

- You can find the entire code of my analysis below.

```
library(tidytext)
library(dplyr)
library(stringr)
library(textreadr)
library(ggplot2)
data(stop_words)


#Custom Stop Words
cusstop <- tribble( ~word, ~lexicon,
                    "wanna", "CUSTOM",
                    "whoa" , "CUSTOM",
                    'gonna', "CUSTOM",
                    'gotta', "CUSTOM",
                    "bum"  , "CUSTOM",
                    "na"   , "CUSTOM",
                    "eh"   , "CUSTOM",
                    "ooh"  , "CUSTOM",
                    'hey'  , "CUSTOM",
                    'la'   , "CUSTOM",
                    'ah'   , "CUSTOM",
                    'uh'   , "CUSTOM",
                    'muh'  , "CUSTOM",
                    'yeah' , 'CUSTOM'
)
stop_words <- stop_words %>% bind_rows(cusstop)



#musician 1 = bieber
my_bieber <- read_document(file="bieber.txt")
my_df1<-data_frame(line=1:3715,text=my_bieber)

my_df1_non <- my_df1 %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words)


#musician 2 = rihanna
my_rihanna <- read_document(file="rihanna.txt")
my_df2<-data_frame(line=1:3895,text=my_rihanna)
```

```
my_df2_non <- my_df2 %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words)

#musician 3 = lady gaga
my_lady <- read_document(file="ladygaga.txt")
my_df3<-data_frame(line=1:3807,text=my_lady)

my_df3_non <- my_df3 %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words)

#musician 4 = drake
my_drake <- read_document(file="drake.txt")
my_df4<-data_frame(line=1:4773,text=my_drake)

my_df4_non <- my_df4 %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words)

#musician 5 = bruno mars
my_brunomars <- read_document(file="brunomars.txt")
my_df5<-data_frame(line=1:3270,text=my_brunomars)

my_df5_non <- my_df5 %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words)



library(tidyr)
frequency <- bind_rows(mutate(my_df1_non, musician="Justin Bieber"),
                        mutate(my_df2_non, musician= "Rihanna"),
                        mutate(my_df3_non, musician="Lady Gaga"),
                        mutate(my_df4_non, musician="Drake"),
                        mutate(my_df5_non, musician="Bruno Mars")
)%>%#closing bind_rows
  mutate(word=str_extract(word, "[a-z']+")) %>%
  count(musician, word) %>%
  group_by(musician) %>%
  mutate(proportion = n/sum(n))%>%
  select(-n) %>%
  spread(musician, proportion) %>%
  gather(musician, proportion, `Lady Gaga`, `Rihanna`,`Drake`,`Bruno Mars`)

print(frequency)
```

```
#let's plot the correlograms:
library(scales)
ggplot(frequency, aes(x=proportion, y=`Justin Bieber`,
                      color = abs(`Justin Bieber`- 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(~musician, ncol=2)+
  theme(legend.position = "none")+
  labs(y= "Justin Bieber", x=NULL)

cor.test(data=frequency[frequency$musician == "Lady Gaga",],
         ~proportion + `Justin Bieber`)

cor.test(data=frequency[frequency$musician == "Drake",],
         ~proportion + `Justin Bieber`)

cor.test(data=frequency[frequency$musician == "Bruno Mars",],
         ~proportion + `Justin Bieber`)

cor.test(data=frequency[frequency$musician == "Rihanna",],
         ~proportion + `Justin Bieber`)

##############################################
# Frequency of Total Data

word_counts <- bind_rows(mutate(my_df1, musician="Justin Bieber"),
                         mutate(my_df2, musician= "Rihanna"),
                         mutate(my_df3, musician="Lady Gaga"),
                         mutate(my_df4, musician="Drake"),
                         mutate(my_df5, musician="Bruno Mars")
)

tidy_pop <- word_counts %>%
  group_by(musician) %>%
  mutate(linenumber = row_number()) %>%
  ungroup() %>%
  unnest_tokens(word, text)%>%
  anti_join(stop_words)
tidy_pop
####### Table #########
```

```
#We can see who is more repeatetive and who used which words often
whousedwhat<-tidy_pop %>%
    filter(n>150)
whousedwhat
####### Table #########


word_co <- word_counts %>%
  mutate(linenumber = row_number()) %>%
  unnest_tokens(word, text)%>%
  anti_join(stop_words)

word_counts2<-word_co%>%
  count(word) %>%
  filter(n>150)%>%
  arrange(desc(n))

word_counts <- word_co %>%
  count(word) %>%
  filter(n>200)%>%
  mutate(word2= fct_reorder(word, n))



##Review Word Counts BAR PLOT
ggplot(
  word_counts, aes(x = word2, y = n)
)+
  geom_col() +
  coord_flip() + ggtitle("Review Word Counts")


#See what sentiments doing
get_sentiments("bing") #negative-positive
get_sentiments("nrc") #feelings trust fear etc.
get_sentiments("afinn") #numbers


#positive-negative thing
library(tidyr)
popsentiment <- tidy_pop %>%
  inner_join(get_sentiments("bing")) %>%
  count(musician,index = linenumber %/% 80, sentiment) %>%
  spread(sentiment, n, fill = 0) %>%
  mutate(sentiment = positive - negative)
```

```
#Put them on a plot
library(ggplot2)
ggplot(popsentiment, aes(index, sentiment, fill = musician)) + geom_col(show.legend = FALSE)
+
  facet_wrap(~musician, ncol = 2, scales = "free_x")


#################################################################3

### Drake
Drake <- tidy_pop %>%
  filter(musician == "Drake")
Drake

#afinn value
afinn <- Drake %>%
  inner_join(get_sentiments("afinn")) %>%
  group_by(index = linenumber %/% 80) %>%
  summarise(sentiment = sum(value)) %>%
  mutate(method = "AFINN")
afinn
#bing and nrc
bing_and_nrc <- bind_rows(
  Drake %>%
    inner_join(get_sentiments("bing")) %>%
    mutate(method = "Bing et al."),
  Drake %>%
    inner_join(get_sentiments("nrc") %>%
                 filter(sentiment %in% c("positive", "negative"))) %>%
    mutate(method = "NRC")) %>%
  count(method, index = linenumber %/% 80, sentiment) %>%
  spread(sentiment, n, fill = 0) %>%
  mutate(sentiment = positive - negative)
bing_and_nrc
#Comparrison afinn and bing&nrc
bind_rows(afinn,
          bing_and_nrc) %>%
  ggplot(aes(index, sentiment, fill = method)) + geom_col(show.legend = FALSE) +
facet_wrap(~method, ncol = 1, scales = "free_y")

#Counting sentiments
get_sentiments("nrc") %>%
  filter(sentiment %in% c("positive", "negative")) %>%
  count(sentiment)

get_sentiments("bing") %>%
  count(sentiment)

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

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()
#######################################################################################
######


#######################################################################################
######

# Word Cloud just for Justin Bieber and Drake

word_c <- bind_rows(mutate(my_df1, musician="Justin Bieber"),
                    mutate(my_df4, musician="Drake")

)


drake_justin <- word_c %>%
  group_by(musician) %>%
  mutate(linenumber = row_number()) %>%
  ungroup() %>%
  unnest_tokens(word, text)%>%
  anti_join(stop_words)
drake_justin


library(wordcloud)
drake_justin %>%
  anti_join(stop_words) %>%
  count(word) %>%
  with(wordcloud(word, n, max.words = 80))
```

```
library(reshape2)
drake_justin %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = TRUE) %>%
  acast(word ~ sentiment, value.var = "n", fill = 0) %>% comparison.cloud(colors = c("gray30",
"gray70"),
                                                                max.words = 80)
```

```
#Wordcloud for all the top 5 musician
library(wordcloud)
tidy_pop %>%
  anti_join(stop_words) %>%
  count(word) %>%
  with(wordcloud(word, n, max.words = 100))

library(reshape2)
tidy_pop %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = TRUE) %>%
  acast(word ~ sentiment, value.var = "n", fill = 0) %>% comparison.cloud(colors = c("gray20",
"gray80"),
                                                                max.words = 100)
```

```
bingnegative <- get_sentiments("bing") %>%
  filter(sentiment == "negative")
wordcounts3 <- tidy_pop %>%
  group_by(musician) %>%
  summarize(words = n())
wordcounts3
```