

# Sentiment Analysis 2

Christine Yuan

## Setup

```
library(readr)
library(tidytext)
library(textdata)
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(ggnewscale)
dataframe <- read_csv("twitch-chat-2264691725.csv", show_col_types = FALSE)
messages <- dataframe[, 4]
unlisted_messages <- unlist(messages)
vector_split_messages <- unlist(strsplit(unlisted_messages, split = " "))
split_messages <- data.frame(vector_split_messages)
names(split_messages) <- "Word"
```

## NRC Custom Lexicon, Top 150 Most Used Words in Twitch Chat with Sentiment

```
nrc_sentiment <- get_sentiments(lexicon = "nrc")
nrc_lexicon <- data.frame("Word" = nrc_sentiment$word, "Sentiment" = nrc_sentiment$sentiment)
nrc_lexicon <- nrc_lexicon[nrc_lexicon$Sentiment == "positive" | nrc_lexicon$Sentiment == "negative", ]

nrc_twitch_words <- c("xdd", "Segz", "deal", "Dance", "o7", "widetime", "no", "widetime3", "widetime2",
nrc_twitch_words_sentiment <- c("laugh at", "movement emote", "positive emote", "positive emote", "laugh")
nrc_twitch_lexicon <- data.frame("Word" = nrc_twitch_words, "Sentiment" = nrc_twitch_words_sentiment)

nrc_new_lexicon <- rbind(nrc_lexicon, nrc_twitch_lexicon)
```

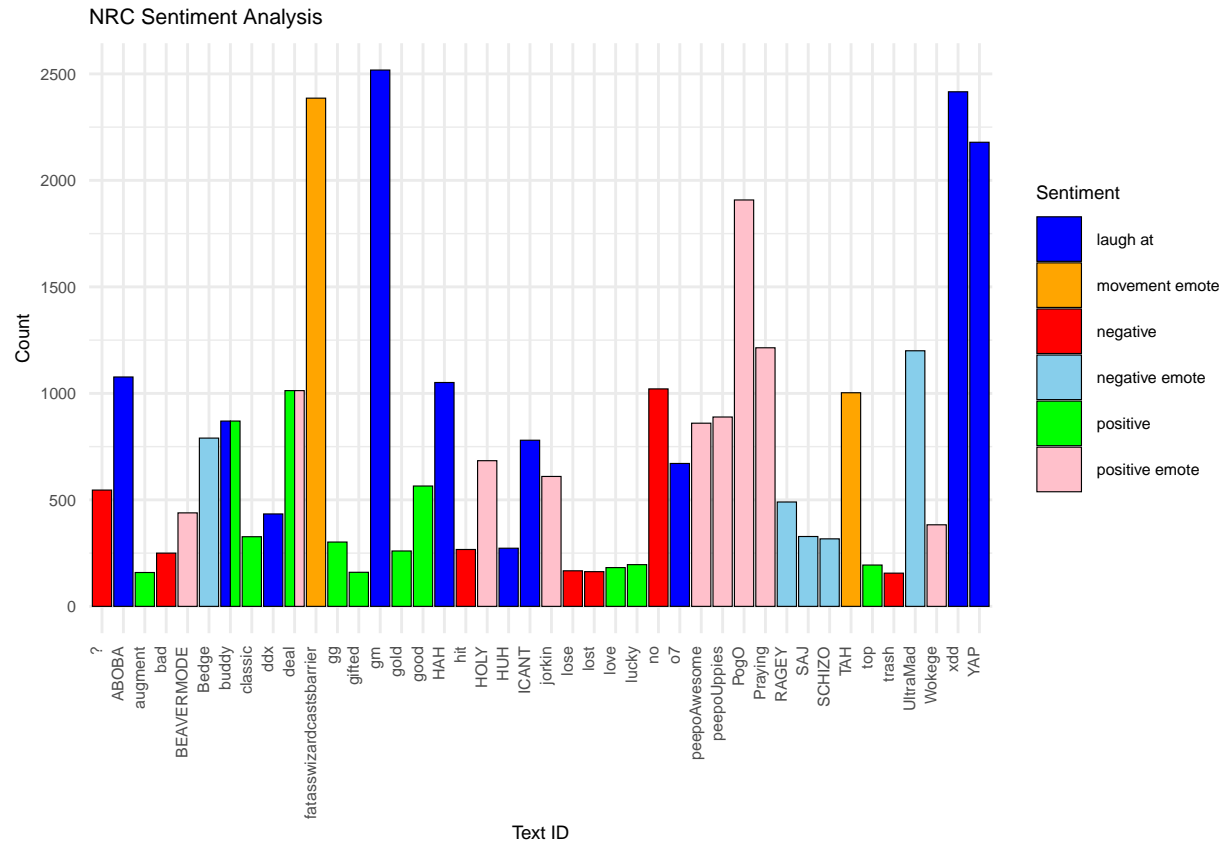
## NRC Sentiment Analysis Graph

```
# Join with sentiment lexicon to classify words
nrc_sentiment <- split_messages %>%
  inner_join(nrc_new_lexicon, by = "Word") %>%
  count(Word, Sentiment, sort = TRUE) %>%
  ungroup()

## Warning in inner_join(., nrc_new_lexicon, by = "Word"): Detected an unexpected many-to-many relationship.
## i Row 16 of 'x' matches multiple rows in 'y'.
## i Row 661 of 'y' matches multiple rows in 'x'.
## i If a many-to-many relationship is expected, set 'relationship =
##   "many-to-many"' to silence this warning.

# Filter
filter_nrc_sentiment <- nrc_sentiment[nrc_sentiment$n >= 150, ]

# Plot the sentiment analysis results
filter_nrc_sentiment %>%
  ggplot(aes(x = factor(Word), y = n, fill = Sentiment)) +
  geom_col(position = "dodge", color = "black", linewidth = 0.2) +
  labs(title = "NRC Sentiment Analysis", x = "Text ID", y = "Count") +
  scale_fill_manual(values = c("positive" = "green", "negative" = "red", "movement emote" = "orange", "laugh at" = "blue")) +
  theme_minimal() +
  theme(text = element_text(size = 7),
        axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0))
```



## Flat Sentiment Analysis Graph (What happens if we don't create a custom lexicon?)

```
flat_nrc_messages <- semi_join(split_messages, nrc_lexicon)
```

```
## Joining with 'by = join_by(Word)'
```

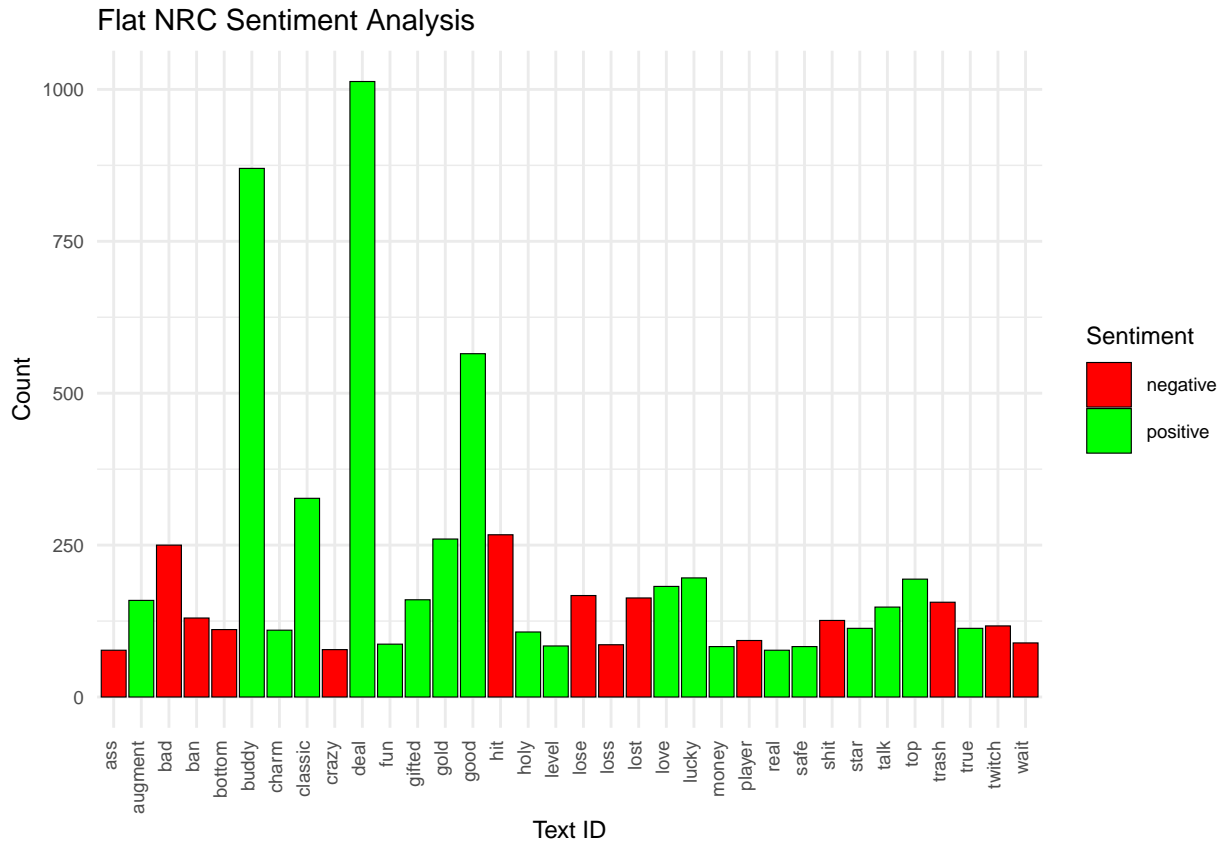
```
flat_nrc_sentiment <- split_messages %>%  
  inner_join(nrc_lexicon, by = "Word") %>%  
  count(Word, Sentiment, sort = TRUE) %>%  
  ungroup()
```

```
## Warning in inner_join(., nrc_lexicon, by = "Word"): Detected an unexpected many-to-many relationship  
## i Row 74 of 'x' matches multiple rows in 'y'.  
## i Row 661 of 'y' matches multiple rows in 'x'.  
## i If a many-to-many relationship is expected, set 'relationship =  
##   "many-to-many"' to silence this warning.
```

```
filter_flat_nrc_sentiment <- flat_nrc_sentiment[flat_nrc_sentiment$Sentiment == "negative" | flat_nrc_s  
small_flat_nrc_sentiment <- filter_flat_nrc_sentiment[-c(1, 2), ]
```

```
# Plot the sentiment analysis results
```

```
filter_flat_nrc_sentiment[filter_flat_nrc_sentiment$n >= 75,] %>%  
  ggplot(aes(x = factor(Word), y = n, fill = Sentiment)) +  
  geom_col(position = "dodge", color = "black", linewidth = 0.2) +  
  labs(title = "Flat NRC Sentiment Analysis", x = "Text ID", y = "Count") +  
  scale_fill_manual(values = c("positive" = "green", "negative" = "red")) +  
  theme_minimal() +  
  theme(text = element_text(size = 9),  
        axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.2))
```



## Afinn Custom Lexicon

```
afinn_sentiment <- get_sentiments(lexicon = "afinn")
afinn_lexicon <- data.frame("Word" = afinn_sentiment$word, "Sentiment" = afinn_sentiment$value)

afinn_twitch_words <- c("xdd", "Segz", "deal", "Dance", "o7", "widetime", "widetime3", "widetime2", "vi
afinn_twitch_words_sentiment <- c(2, 1, 3, 3, 2, 1, 1, 1, 3, 2, 2, 1, 2, 3, 2, -4, 1, 2, 1, 1, -4, -3, 1
afinn_twitch_lexicon <- data.frame("Word" = afinn_twitch_words, "Sentiment" = afinn_twitch_words_sentiment)

afinn_new_lexicon <- rbind(afinn_lexicon, afinn_twitch_lexicon)
```

## Afinn Sentiment Analysis Graph

```
# Join with sentiment lexicon to classify words
afinn_sentiment <- split_messages %>%
  inner_join(afinn_new_lexicon, by = "Word") %>%
  count(Word, Sentiment, sort = TRUE) %>%
  ungroup()

## Warning in inner_join(., afinn_new_lexicon, by = "Word"): Detected an unexpected many-to-many relationship.
## i Row 28458 of 'x' matches multiple rows in 'y'.
## i Row 2487 of 'y' matches multiple rows in 'x'.
## i If a many-to-many relationship is expected, set 'relationship =
##   "many-to-many"' to silence this warning.

# Filter
filter_afinn_sentiment <- afinn_sentiment[afinn_sentiment$n >= 200, ]
filter_afinn_sentiment$Sentiment[filter_afinn_sentiment$Word == "no"] <- -4

# Plot the sentiment analysis results
filter_afinn_sentiment %>%
  ggplot(mapping = aes(x = Word, y = Sentiment, fill = n)) +
  geom_col(color = "black", linewidth = 0.2) +
  theme_minimal() +
  theme(text = element_text(size = 7),
        axis.text.x = element_text(angle = 90, hjust = 1, vjust = -0.2)) +
  scale_y_continuous(label = seq(-5, 5), breaks = seq(-5, 5)) +
  labs(title = "Afinn Sentiment Analysis", x = "Text ID", y = "Sentiment Value", fill = "Frequency") +
  scale_fill_gradient(low = "white", high = "purple")
```

