```
# Student: Anthony J Buchanan, Mentor: Raghunandan Patthar
# File Description: Reads in data on Reddit comments and does sentiment analysis

# read the comments data into the comments data frame
 devdb <- dbConnect(RSQLServer::SQLServer(), server="localhost", port=1433,
                    properties=list(user="rdata", password="password"))

 comments_raw <- dbGetQuery(devdb, " select top 30000 score_category, body,
                 subreddit from Comments where author not like '%bot%' and
                 author not like '%moderator%' and body not in
                 (select body from Comments group by body having count(*) > 1)")
  #comments_raw <- read.csv("comments_sent.csv", stringsAsFactors = FALSE)
  names(comments_raw)[1] <- "score_category" #rename because csv is off
  names(comments_raw)[2] <- "body"
  names(comments_raw)[3] <- "subreddit"

> # examine the structure of the comments data
'data.frame':  30000 obs. of  2 variables:
 $ score_category: chr  "Neutral" "Neutral" "Negative" "Positive" ...
 $ body          : chr  "I'm new to basketball trades and stuff, but is it just a small g

> table(comments_raw$score_category)

Negative  Neutral Positive
    7233    12516    10251

> comments_dtm <- DocumentTermMatrix(corpus_clean)

<<DocumentTermMatrix (documents: 30000, terms: 51364)>>
Non-/sparse entries: 417543/1540502457
Sparsity           : 100%
Maximal term length: 267
Weighting          : term frequency (tf)

> # creating training and test datasets
> comments_raw_train <- comments_raw[1:22500, ]
> comments_raw_test  <- comments_raw[22501:30000, ]
> comments_dtm_train <- comments_dtm[1:22500, ]
> comments_dtm_test  <- comments_dtm[22501:30000, ]
> comments_corpus_train <- corpus_clean[1:22500]
> comments_corpus_test  <- corpus_clean[22501:30000]

> # check that the proportion of score category is similar
> prop.table(table(comments_raw_train$score_category))

Negative   Neutral  Positive
0.2435556 0.4138222 0.3426222

> prop.table(table(comments_raw_test$score_category))

Negative   Neutral  Positive
0.2337333 0.4273333 0.3389333

> # word cloud visualization
> wordcloud(comments_corpus_train, min.freq = 30, random.order = FALSE)
> wordcloud(positive$body, max.words = 40, scale = c(3, 0.5))
> wordcloud(neutral$body, max.words = 40, scale = c(3, 0.5))
> wordcloud(negative$body, max.words = 40, scale = c(3, 0.5))
```

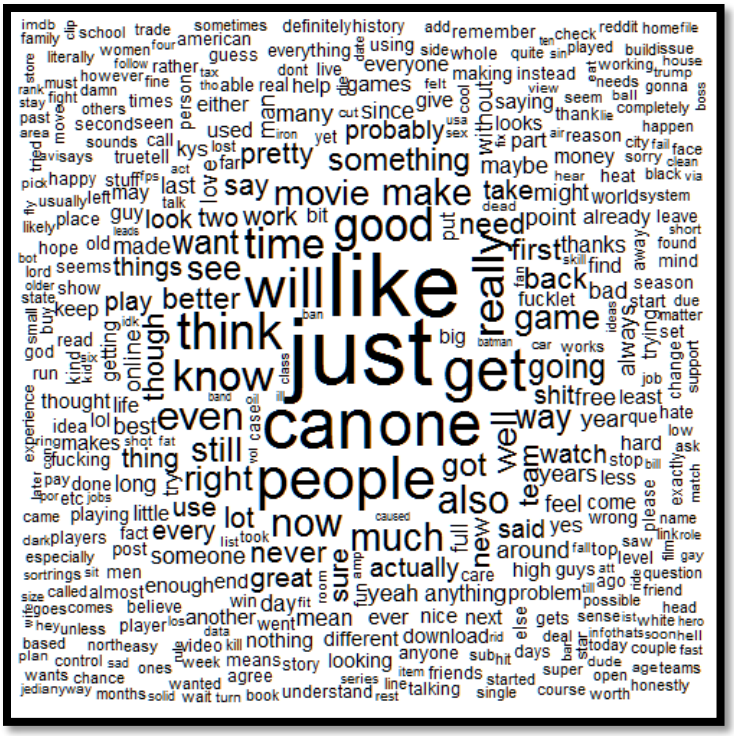

Figure 1 Training Data (All)

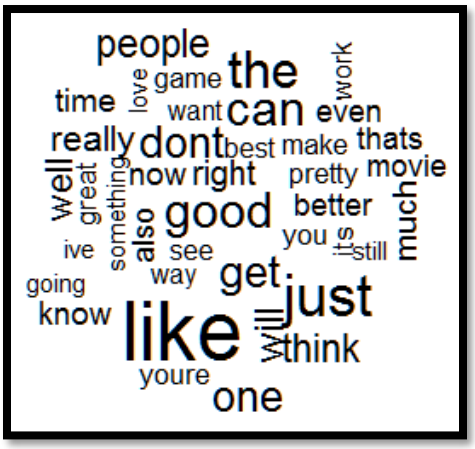

Figure 2 Training Data (Positive)

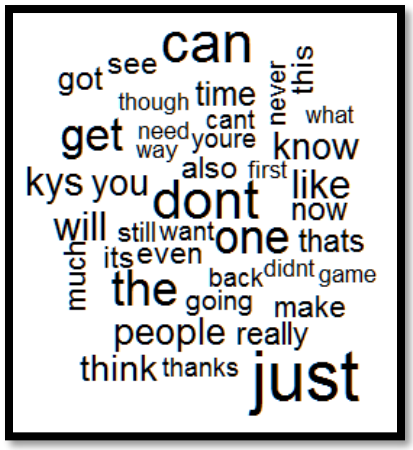

Figure 3 Training Data (Neutral)

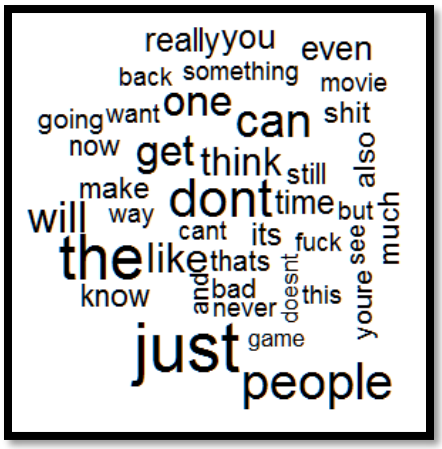

Figure 4 Training Data (Negative)

Evaluating model performance ----

Cell Contents
|-------------------------|
| N |
N / Col Total

 Total Observations in Table: 7500

| actual
predicted	Negative	Neutral	Positive	Row Total
Negative	642	186	153	981
	0.366	0.058	0.060	
Neutral	921	2726	1273	4920
	0.525	0.851	0.501	
Positive	190	293	1116	1599
	0.108	0.091	0.439	
Column Total	1753	3205	2542	7500
	0.234	0.427	0.339	

Improving model performance ----

Cell Contents
|-------------------------|
| N |
N / Col Total

Total Observations in Table: 7500

| actual
predicted	Negative	Neutral	Positive	Row Total
Negative	603	153	123	879
	0.344	0.048	0.048	
Neutral	957	2762	1328	5047
	0.546	0.862	0.522	
Positive	193	290	1091	1574
	0.110	0.090	0.429	
Column Total	1753	3205	2542	7500
	0.234	0.427	0.339	