install.packages('plyr')

library(plyr)

install.packages('stringr')

library(stringr)

#my data

datafile <- file.choose()

data <- read.csv(datafile)

#reading data for positive and negative words

positivefile <- file.choose()

positive\_words <- read.delim("positive-words.txt")

negativefile <- file.choose()

negative\_words <- read.delim("negative-words.txt")

#getting words

positive\_words <- scan("positive-words.txt", what = 'character')

negative\_words <- scan("negative-words.txt", what = 'character')

negative\_words <- c(negative\_words, 'wtf', 'shitty', 'sucks')

#sentiment analysis

sentimentScore <- function(tweets, positive\_words, negative\_words, .progress='none')

{

require(plyr)

require(stringr)

scores = laply(sentences, function(tweets, positive\_words, negative\_words) {

# cleaning

tweets = gsub('[[:punct:]]', '', tweets)

tweets = gsub('[[:cntrl:]]', '', tweets)

tweets = gsub('\\d+', '', tweets)

# and convert to lower case:

tweets = tolower(tweets)

# split into words. str\_split is in the stringr package

word.list = str\_split(tweets, '\\s+')

# sometimes a list() is one level of hierarchy too much

words = unlist(word.list)

# compare our words to the dictionaries of positive & negative terms

positive.matches = match(words, positive\_words)

negative.matches = match(words, negative\_words)

# match() returns the position of the matched term or NA

# we just want a TRUE/FALSE:

positive.matches = !is.na(positive.matches)

negative.matches = !is.na(negative.matches)

# and conveniently enough, TRUE/FALSE will be treated as 1/0 by sum():

score = sum(positive.matches) - sum(negative.matches)

return(score)

}, positive\_words, negative\_words, .progress=.progress )

scores.df = data.frame(score=scores, text=sentences)

return(scores.df)

}

result <- sentimentScore(data$text,positive\_words,negative\_words)

summary(result$score)

hist(result$score, main = "Histogram for Tweets", ylab = "Count of Tweets", border = "blue", col = "green", ylim = c(0,500))