DATA MINING PROJECT-1

TEXT MINING using Twitter Data



Ayberk Ece ÖRTEN B1505.090046

Hatice MENTEŞE B1505.090013

**Source Code(in R)**

**```{r}**

**#import libraries**

**library(plyr) #to split and combine data**

**library(stringr) #string operations**

**```**

**Add a new chunk by clicking the \*Insert Chunk\* button on the toolbar or by pressing \*Ctrl+Alt+I\*.**

**```{r}**

**#reading data for positive and negative words**

**positivefile <- file.choose()**

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

**```**

**```{r}**

**negativefile <- file.choose()**

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

**```**

**```{r}**

**#getting words**

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

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

**```**

**```{r}**

**#declaring pozitive and negative words**

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

**positive\_words <- c(positive\_words, 'amazing', 'good', 'perfect')**

**```**

**```{r}**

**#sentiment analysis (deciding whether its positive or negative)**

**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)**

**}**

**```**

**```{r}**

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

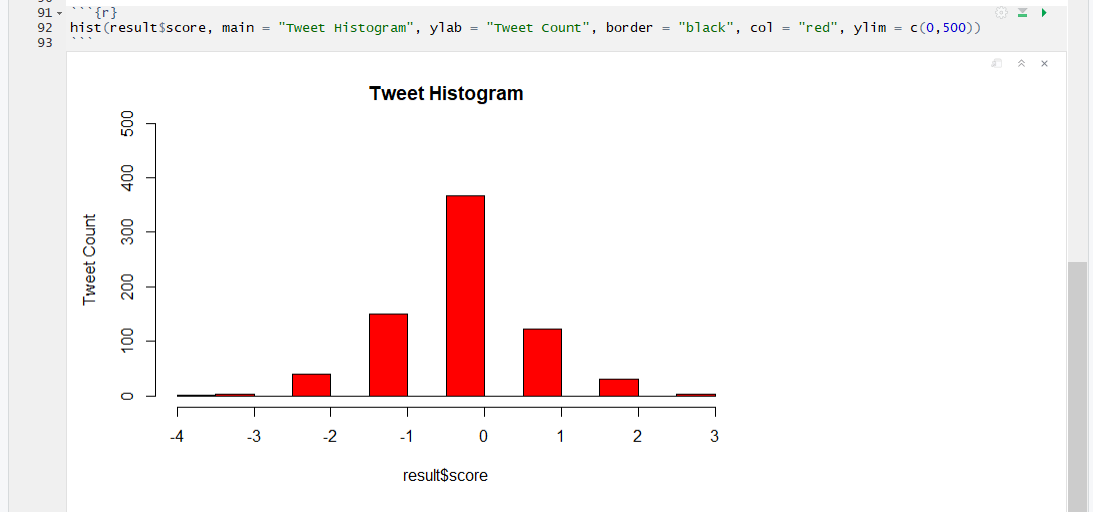
**```**

**```{r}**

**summary(result$score)**

**```**

**Output:**



**In this histogram, tweets aree divided by positivity and negativity.**

**Output2:**

