SENTIMENT ANALYSIS IN SENTENCES

A PROJECT DOCUMENT

Submitted by

VYSAKH S

SENTIMENT ANALYSIS IN SENTENCES

ABSTRACT:

Sentiments have become major part in our life. People are more concern about the sentiments in the texts or sentences in the message in social medias and everywhere. Sentiment analysis is a vital area of natural language processing (NLP) that aims to automatically identify and categorize the sentiment expressed in textual data. The concept of analysis involves in removing the unwanted words like stop words, punctuations, numbers etc… Only after removing those we can analyze the sentiments in sentences.

PROJECT DESCRIPTION:

The objective of this project is to develop a sentiment analysis model in R that can classify the sentiments expressed in sentences. Sentiment analysis uses natural language processing (NLP), Text mining, and some methods to find the sentiments in the sentences that involves identifying and categorizing the sentiment (positive, negative, or neutral) conveyed in textual data.

Also we used the NRC emotion lexicon is a list of English words and their associations with eight basic emotions (anger, fear, anticipation, trust, surprise, sadness, joy, and disgust) and two sentiments (negative and positive)

The sentiment scores are found using certain methods like syuzhet, bing, afinn and nrc. But we use mostly syuzhet score for further score calculation. The sentiment score is calculated only after cleaning and transforming data's according to what the model is needed.

Bar plot and pie chart is been used to visualize various types of sentiments in group of sentences.

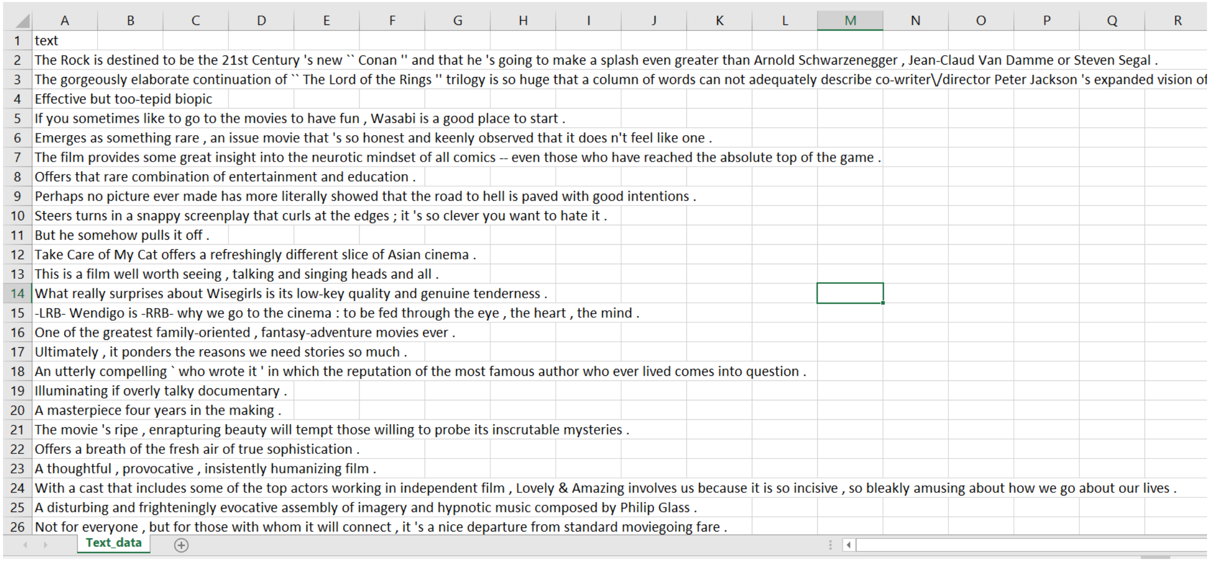
REQUIREMENTS:

We have to first select the dataset with sentences and install some dependencies in RStudio:

```
library(NLP)
library(tm)
library(RColorBrewer)
library(SnowballC)
library(stringr)
library(syuzhet)
```

DATASET:

The dataset that I have collected has number of sentences stored in csv format (it can also be stored in excel format)



	A
1	text
2	The Rock is destined to be the 21st Century 's new `` Conan '' and that he 's going to make a splash even greater than Arnold Schwarzenegger , Jean-Claud Van Damme or Steven Segal .
3	The gorgeously elaborate continuation of `` The Lord of the Rings '' trilogy is so huge that a column of words can not adequately describe co-writer\/director Peter Jackson 's expanded vision of
4	Effective but too-tepid biopic
5	If you sometimes like to go to the movies to have fun , Wasabi is a good place to start .
6	Emerges as something rare , an issue movie that 's so honest and keenly observed that it does n't feel like one .
7	The film provides some great insight into the neurotic mindset of all comics -- even those who have reached the absolute top of the game .
8	Offers that rare combination of entertainment and education .
9	Perhaps no picture ever made has more literally showed that the road to hell is paved with good intentions .
10	Steers turns in a snappy screenplay that curls at the edges ; it 's so clever you want to hate it .
11	But he somehow pulls it off .
12	Take Care of My Cat offers a refreshingly different slice of Asian cinema .
13	This is a film well worth seeing , talking and singing heads and all .
14	What really surprises about Wisegirls is its low-key quality and genuine tenderness .
15	-LRB- Wendigo is -RRB- why we go to the cinema : to be fed through the eye , the heart , the mind .
16	One of the greatest family-oriented , fantasy-adventure movies ever .
17	Ultimately , it ponders the reasons we need stories so much .
18	An utterly compelling ` who wrote it ' in which the reputation of the most famous author who ever lived comes into question .
19	Illuminating if overly talky documentary .
20	A masterpiece four years in the making .
21	The movie 's ripe , enrapturing beauty will tempt those willing to probe its inscrutable mysteries .
22	Offers a breath of the fresh air of true sophistication .
23	A thoughtful , provocative , insistently humanizing film .
24	With a cast that includes some of the top actors working in independent film , Lovely & Amazing involves us because it is so incisive , so bleakly amusing about how we go about our lives .
25	A disturbing and frighteningly evocative assembly of imagery and hypnotic music composed by Philip Glass .
26	Not for everyone , but for those with whom it will connect , it 's a nice departure from standard moviegoing fare .

Link for Dataset: Dataset for sentiment analysis performed by 2005054

CODE:

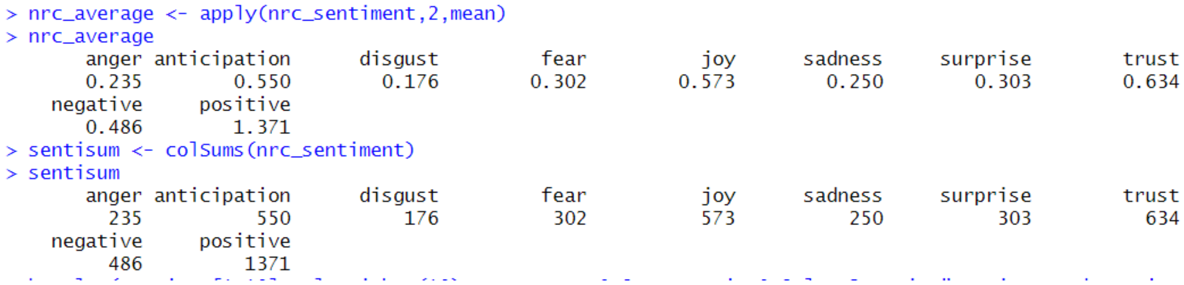
```
# Install required packages if not already installed
install.packages(c("NLP","tm","RColorBrewer","SnowballC","stringr","syuzhet"))

library(NLP)
library(tm)
library(RColorBrewer)
library(SnowballC)
library(stringr)
library(syuzhet)

# read input file
reviews <- read.csv(file.choose(), sep=",", header=T)

abc <- as.matrix(reviews)
head(abc)
tail(abc)

#text data cleaning

# stringr functions for removing symbols
abc <- str_remove_all(abc,"–")
abc <- str_remove_all(abc,"’")
abc <- str_remove_all(abc,"—")

abc <- str_remove_all(abc,"“")

abc <- str_remove_all(abc,"”")


# tm functions for text cleaning

abc<-removeNumbers(abc)

abc<-removePunctuation(abc)

abc<-tolower(abc)

abc<-removeWords(abc,c("now", "one", "will", "may", "says", "said", "also", "figure", "etc", "re", "can"))

stopwords<-c("the", "and", stopwords("en"))

abc<-removeWords(abc, stopwords("en"))

abc<-stripWhitespace(abc) #same as str_trim

abc<-wordStem(abc)       #function from SnowballC


review_text<-abc

head(review_text)

tail(review_text)


# Sentiment analysis:

# sentiment score using get_sentiment() function & scoring method

# scoring mehods: syuzhet, bing, afinn, nrc

# Each method may have different scale
```

```
syuzhet_score <- get_sentiment(review_text, method="syuzhet")
head(syuzhet_score)
summary(syuzhet_score)

bing_score <- get_sentiment(review_text, method="bing")
head(bing_score)
summary(bing_score)

afinn_score <- get_sentiment(review_text, method="afinn")
head(afinn_score)
summary(afinn_score)

nrc_score <- get_sentiment(review_text, method="nrc")
head(nrc_score)
summary(nrc_score)

comb_score <- cbind(syuzhet_score, bing_score, afinn_score, nrc_score)
dimnames(comb_score) <- list(1:nrow(comb_score), c("s1", "s2", "s3", "s4"))
df <- as.data.frame(comb_score)
head(df,20)

# simple analysis based on syuzhet_score
min(df$s1)
```

```
minLoc1 <- which(df$s1==min(df$s1))
minLoc1
review_text[minLoc1]

max(df$s1)
maxLoc1 <- which(df$s1==max(df$s1))
maxLoc1
review_text[maxLoc1]


#view text docs with extreme negative sentiment score
syuz_neg <- which(syuzhet_score<=(-5))
txt<-review_text[syuz_neg]
result<-cbind(syuz_neg,txt)
result

#view text docs with high posive sentiment score
syuz_posit <- which(syuzhet_score>=4.5)
txt<-review_text[syuz_posit]
result1<-cbind(syuz_posit,txt)
result1

# Analysis given above has limitations wrt scale
# scale used by the text mining methods differ

# sentiment score normalized with sign function
# sign function assigns +1 for values > 0
# sign function assigns -1 for values < 0
# sign function assigns 0 for values == 0

norm_score <- cbind(
  sign(syuzhet_score),
  sign(bing_score),
  sign(afinn_score),
  sign(nrc_score))

dimnames(norm_score)<-list(1:nrow(norm_score), c("x1", "x2", "x3", "x4"))
head(norm_score)

z<-as.data.frame(norm_score)
head(z,20)

round(prop.table(table(z$x1)),2)    #syuzhet score

#  Emotion classification & positive and negative sentiments
#  "Emotion classification is built on the
#  NRC Word-Emotion Association Lexicon (aka EmoLex)"
#  "The NRC Emotion Lexicon is a list of English words and
#  their associations with eight basic emotions (anger, fear,
#  anticipation, trust, surprise, sadness, joy, and disgust) and
#  two sentiments (negative and positive)"

nrc_sentiment <- get_nrc_sentiment(review_text)
head(nrc_sentiment,20)
tail(nrc_sentiment,20)

nrc_average <- apply(nrc_sentiment,2,mean)
nrc_average

sentisum <- colSums(nrc_sentiment)
sentisum

barplot(sentisum[1:10],col=rainbow(10), cex.names=0.8, cex.axis=0.8,
       las=2, main="Emotions and Sentiments")

Lb <- paste(names(sentisum), ",", sentisum)
pie(sentisum[1:10],col=brewer.pal(8,'Dark2'), labels=Lb,
    main="Emotions and Sentiment nrc Scores", cex=0.8, cex.main=2)
```

OUTPUT:

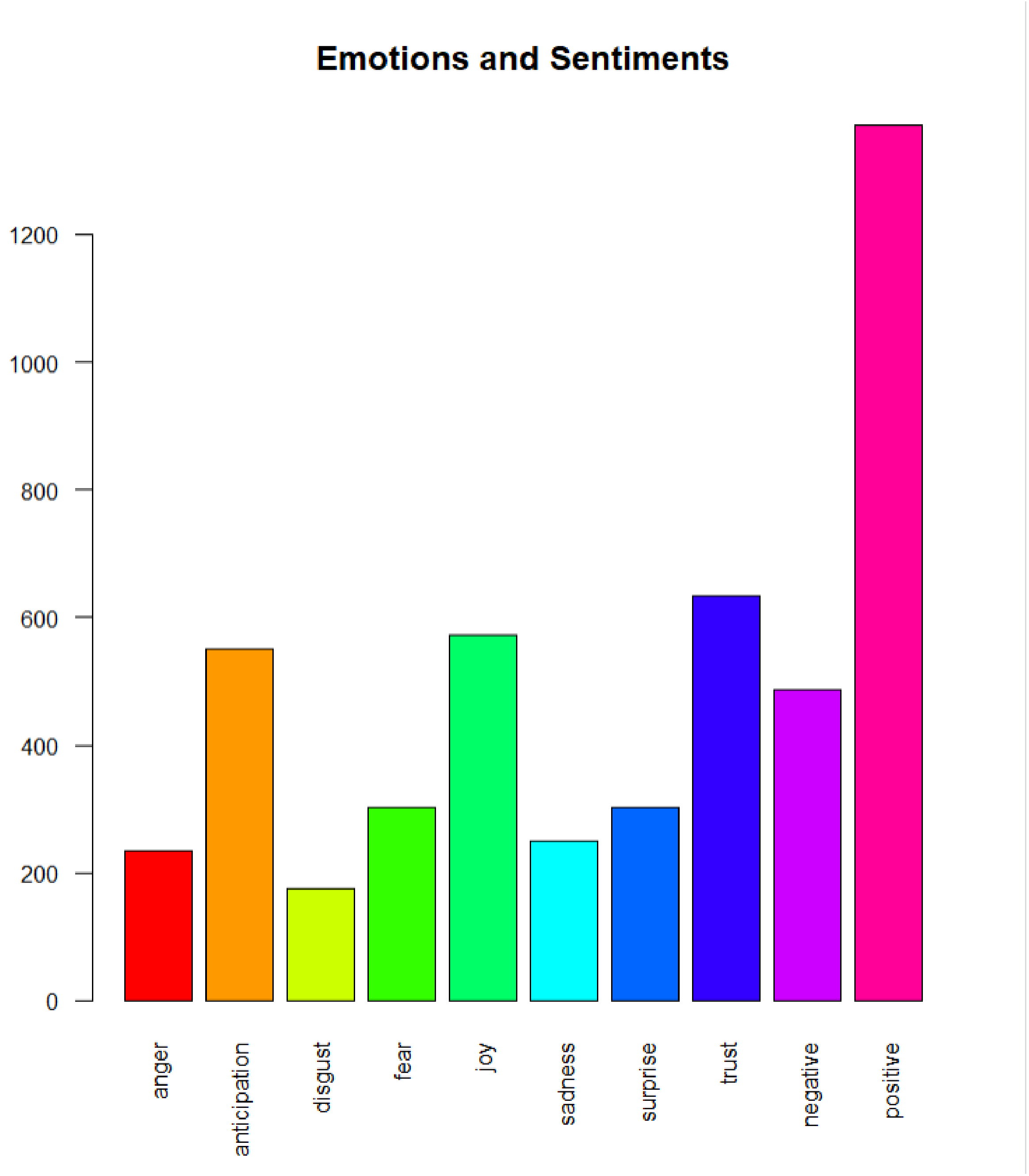
For the given dataset we got output as:



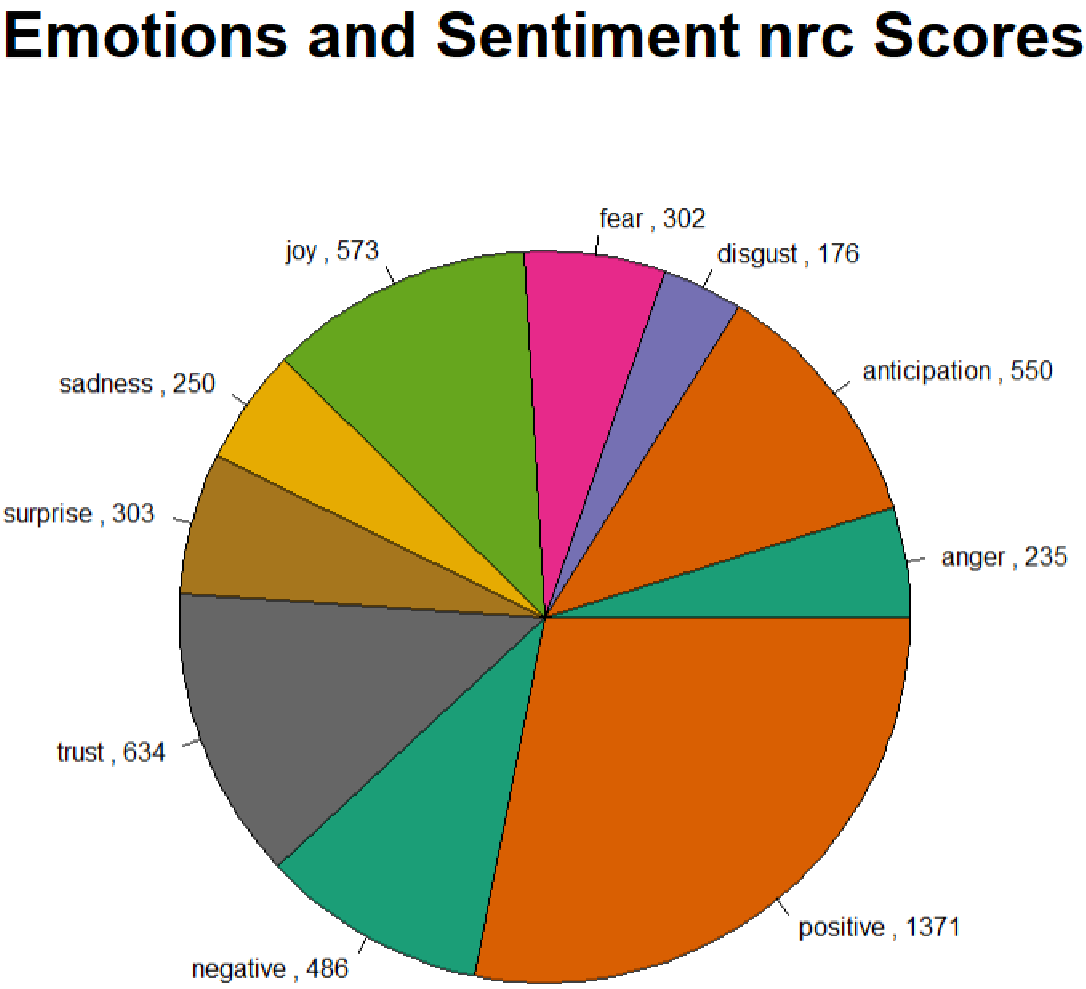
```
> nrc_average <- apply(nrc_sentiment,2,mean)
> nrc_average
      anger anticipation      disgust         fear          joy      sadness     surprise        trust 
      0.235        0.550        0.176        0.302        0.573        0.250        0.303        0.634 
   negative     positive 
      0.486        1.371 
> sentisum <- colSums(nrc_sentiment)
> sentisum
      anger anticipation      disgust         fear          joy      sadness     surprise        trust 
        235          550          176          302          573          250          303          634 
   negative     positive 
        486         1371 
```

VISUALIZATION:

1.BARPLOT:



2.PIE CHART:


Emotions and Sentiment nrc Scores
fear , 302
disgust , 176
joy , 573
anticipation , 550
sadness , 250
surprise , 303
anger , 235
trust , 634
positive , 1371
negative , 486

RESULT & CONCLUSION:

Finally, we got the Sentiment analysis for the sentences in the dataset and made a clear visualization of the sentiments in the sentences.

From the above visualization we can infer that we got:

- ✓ anger - 235
- ✓ anticipation - 550
- ✓ disgust - 176
- ✓ fear - 302
- ✓ joy - 573
- ✓ sadness - 250
- ✓ surprise - 303
- ✓ trust - 634
- ✓ negative - 486
- ✓ positive - 1371

from the dataset that we choose.

Link to get complete R file: R file for the sentiment analysis project