

## CASE STUDY

# Sentiment Analysis

Sentiment analysis uses natural language processing, **text analysis**, and **statistics** to analyze customer sentiment.

Sentiment analysis studies the subjective information in an expression: opinions, appraisals, emotions, or attitudes towards a topic, person or entity.

Expressions can be classified as positive, negative, or neutral.

### Data:

#### Amazon Product Review

Extract reviews using Amazon Reviews Exporter,  
Product: Ikigai (book).

### Output:

**Sentiment scores** based on reviews.

### Dataframe :

Attributes	Type
id	character
profile name	character
text	character
date	character

title	character
rating	integer
images	character
helpful	integer

## Packages :

### 1. Syuzhet -

The package comes with four sentiment dictionaries and provides a method for accessing the robust, but computationally expensive, sentiment extraction tool developed in the NLP group at Stanford.

- **get\_nrc\_sentiment** : Calls the NRC sentiment dictionary to calculate the presence of eight different emotions and their corresponding valence in a text file.

### 2. Wordcloud -

Functionality to create pretty word clouds, visualize differences and similarity between documents, and avoid overplotting in scatter plots with text.

### 3. tm -

A framework for text mining applications within R.

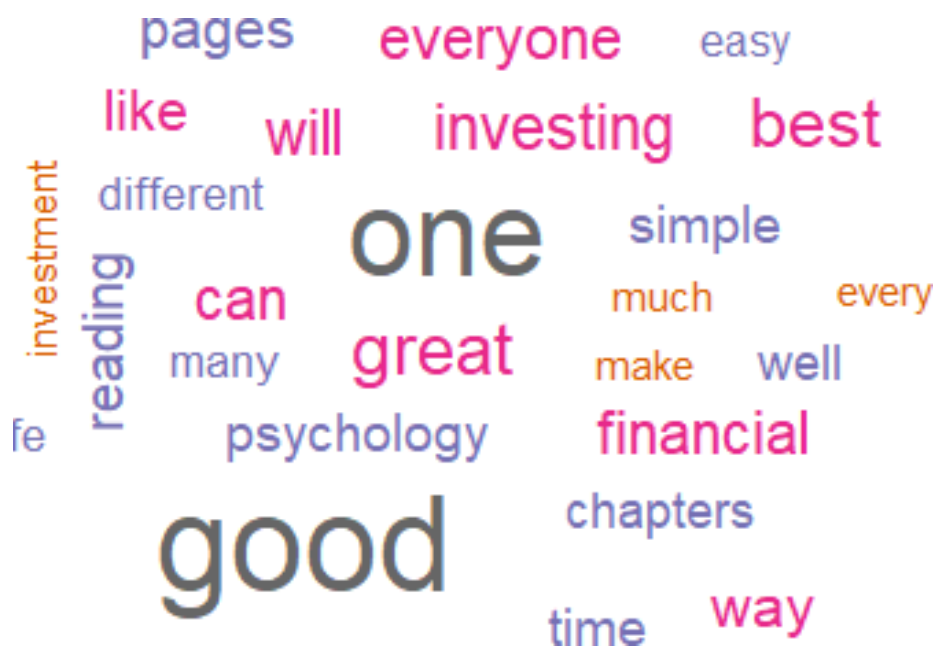
- **corpus** : The main structure for managing documents in tm is a so-called Corpus, representing a collection of text documents. A corpus is an abstract concept, and there can exist several implementations in parallel.
- **tm\_map** : Interface to apply transformation functions (also denoted as mappings) to corpora.

THE TEN EMOTIONS ARE:

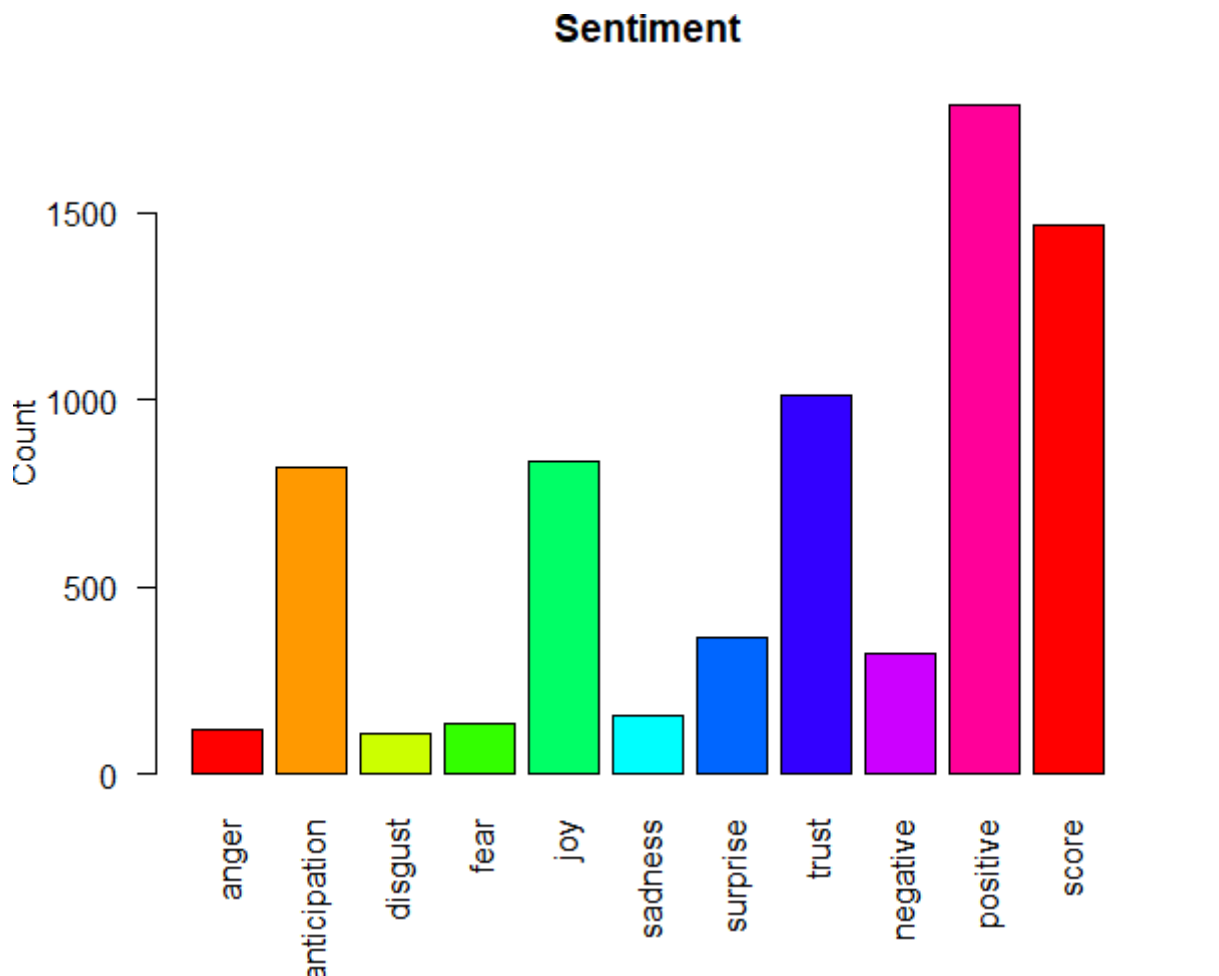
1. anger
2. anticipation
3. disgust
4. fear
5. joy
6. sadness
7. surprise
8. trust
9. negative
10. positive

```
print(review_score)
      anger anticipation    disgust    fear    joy    sadness
      117      821      108    135    837      159
surprise    trust    negative    positive    score
      367    1011    322    1787    1465
```

IMG: SENTIMENT SCORE



IMG: WORD CLOUD



**IMG: RESULT**

### **CONCLUSION:**

The above graph shows that the overall product review is **POSITIVE**.

Similarly we can calculate sentiment scores for other products too and build a NLP model.

### **CODE**

**#For data set of reviews, use Amazon Reviews Exporter to extract reviews**

**#product used : Ikigai (book)**

**#load packages into R**

**library(wordcloud)**                    **#create word cloud**  
**library(tm)**                            **#text mining (text analytics)**  
**library(syuzhet)**                    **# Syuzhet is concerned with the manner in which**  
   **#the elements of the story (fabula)**  
   **#are organized (syuzhet).**

**install.packages("syuzhet")**

**install.packages("wordcloud")**

**install.packages("tm")**

**#import data into R**

**reviews1 <- read.csv(file.choose(), header=T)**

**#to check the structure of file**

**str(reviews1)**

**#use corpus**

**#this function translates value labels into a specific encoding**

**corpus <- iconv(reviews1\$text)**

**corpus <- Corpus(VectorSource(corpus))**

**inspect(corpus[1:5])**                    **#only 1st 5 observations, 1st doc is first review..**

**#cleaning corpus**

**corpus <- tm\_map(corpus, tolower)**                    **#convert to lower case**

```
corpus <-tm_map(corpus,removePunctuation)
```

```
corpus <-tm_map(corpus,removeNumbers)
```

```
corpus <-tm_map(corpus,removeWords,stopwords("english"))
```

```
corpus <-tm_map(corpus,removeWords,c("book","money","read"))
```

```
corpus <-tm_map(corpus,stripWhitespace) #remove spaces
```

```
inspect(corpus[1:5])
```

```
reviews_final <- corpus #creating final object
```

```
#create a TERM DOCUMENT
```

```
#a way of representing the words in the text,
```

```
#as a table (or matrix) of numbers.
```

```
# rows-text responses to be analysed, columns- text that are to be used in  
the analysis.
```

```
tdm <-TermDocumentMatrix(reviews_final)
```

```
tdm <-as.matrix(tdm)
```

```
tdm[1:10,1:5] # 1st 10 rows, 1st 5 columns
```

```
# Docs  
# Terms 1 2 3 4 5  
# aside 1 0 0 0 0
```

```
# book      4 2 0 10 3
# brilliant  1 0 0 0 0
# change     1 0 0 0 0
# define     1 0 0 0 0
# deliver    1 0 0 0 0
# differently 1 0 0 0 0
# even       1 0 0 0 0
# everyone   2 1 0 0 0
# expectations 2 0 0 0 0
```

**#Bar plot of words**

```
w <- rowSums(tdm)
w <- subset(w,w>25)      #only words whose occurrence is greater than 25
barplot(w, las=2, col="blue")
```

**#Create a word cloud**

```
w <- sort(rowSums(tdm),decreasing = T) #T-true
wordcloud(words = names(w),
  freq = w,
  max.words = 40,
  random.order = T,
  min.freq=5,
  colors = brewer.pal(25,"Dark2"),
  scale = c(3,0.3))
```

**#obtain sentiment scores**

```
sentiment_data <- iconv(reviews1$text)
#take text variable pass that into fn get_nrc_sentiment
```

```
s <- get_nrc_sentiment(sentiment_data)
```

```
#which gives us the score on emotions
```

```
s[1:10,]
```

```
#calculate review wise score
```

```
s$score <- s$positive - s$negative
```

```
s[1:10,]
```

```
#write scores into a csv file
```

```
write.csv(x=s, file = "C:/Users/sejal/Desktop/Final_score2.csv" )
```

```
#check product(book) sentiment
```

```
review_score <- colSums(s[,]) #summing columns
```

```
print(review_score)
```

```
#bar plot
```

```
barplot(colSums(s),
```

```
  las=2,
```

```
  col = rainbow(10),
```

```
  ylab = 'Count',
```

```
  main = 'Sentiment')
```

```
#Since positive > negative, overall product review is positive
```



