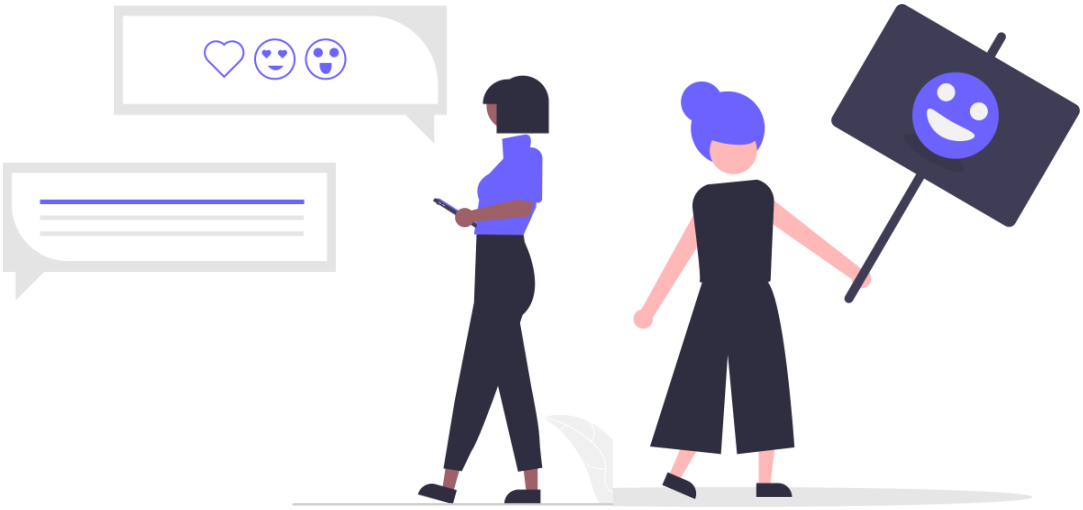
NLP PROJECT

SENTIMENT ANALSYSIS USING R



GURSHEEN KAUR ANAND
ROLL NO : 221
gm.gursheenkaur.anand@gnkhalsa.edu.in

CONTENTS PAGE

INTRODUCTION

We, humans, communicate with each other in a variety of languages, and any language is just a mediator or a way in which we try to express ourselves. And, whatever we say has a sentiment associated with it. It might be positive or negative or it might be neutral as well. Sentiment analysis (or opinion mining) is a natural language processing(NLP) technique used to determine whether data is positive, negative, or neutral. Sentiment analysis is often performed on textual data to help businesses monitor brand and product sentiment in customer feedback, and understand customer needs.

Sentiment analysis focuses on the polarity of a text (positive, negative, neutral) but it also goes beyond polarity to detect specific feelings and emotions (angry, happy, sad, etc), urgency (urgent, not urgent), and even intentions (interested v. not interested). Depending on how you want to interpret customer feedback and queries, you can define and tailor your categories to meet your sentiment analysis needs.

Example
Suppose, there is a fast-food chain company and they sell a variety of different food items like burgers, pizza, sandwiches, milkshakes, etc. They have created a website to sell their food and now the customers can order any food item from their website and they can provide reviews as well, like whether they liked the food or hated it.

- User Review 1: I love this cheese sandwich, it's so delicious.
- User Review 2: This chicken burger has a very bad taste.
- User Review 3: I ordered this pizza today.

So, as we can see that out of these above 3 reviews,

- The first review is definitely a positive one and it signifies that the customer was really happy with the sandwich.
- The second review is negative, and hence the company needs to look into their burger department.
- And, the third one doesn't signify whether that customer is happy or not, and hence we can consider this as a neutral statement.

By looking at the above reviews, the company can now conclude, that it needs to focus more on the production and promotion of their sandwiches as well as improve the quality of their burgers if they want to increase their overall sales.

ABSTRACT

Twitter Sentiment Analysis System is an application that uses a dataset containing various tweets from users about airline organizations. The analysis helps to study Twitter comments and tries to predict whether a review is positive or negative. The dataset contains more than 500K reviews with a number of upvotes & total votes to those comments. These tweets can be used to analyze and categorize the data in such a way that the average sentiment of the public can be found. The derived sentiment can be used for Quality of Service evaluations which will, in turn, be used for improving the performance of the various departments of the organization from the root level.

The application consists of general steps including -

Installing Dependencies + Preparations
Loading the dataset
Sentiment Analysis
Visualization of the results
Analyzing the results and plots

Each of the above steps is intertwined with the other and serves a specific purpose in order to get the desired results.

Why R is used?
R is an integrated suite of software facilities for data manipulation, calculation, and graphical display. It includes

- an effective data handling and storage facility,
- a large, coherent, integrated collection of intermediate tools for data analysis,
- graphical facilities for data analysis and display either on-screen or on hardcopy

The application involves handling large datasets and the required output should contain a visualization graph or any such representation which will make it more readable. Since both of these requirements are successfully fulfilled by R, we use the R environment to create and execute the code. R's capabilities are extended through user-created packages, which offer statistical techniques, graphical devices, import/export, reporting, etc. R's packages and the ease of installing and using them have been cited as driving the language's widespread adoption in data science.

RELATED WORK

Sentiment analysis focuses on the polarity of a text (positive, negative, neutral) but it also goes beyond polarity to detect specific feelings and emotions (angry, happy, sad, etc), urgency (urgent, not urgent), and even intentions (interested v. not interested). Some prominent works in this area include -

Graded Sentiment Analysis

If polarity precision is important to your business, you might consider expanding your polarity categories to include different levels of positive and negative:

- Very positive
- Positive
- Neutral
- Negative
- Very negative

This is usually referred to as graded or fine-grained sentiment analysis, and could be used to interpret 5-star ratings in a review, for example:

1. Very Positive = 5 stars
2. Very Negative = 1 star

Emotion detection

Emotion detection sentiment analysis allows you to go beyond polarity to detect emotions, like happiness, frustration, anger, and sadness. Many emotion detection systems use lexicons (i.e. lists of words and the emotions they convey) or complex machine learning algorithms.

RELATED WORK

Aspect-based Sentiment Analysis

Usually, when analyzing sentiments of texts you'll want to know which particular aspects or features people are mentioning in a positive, neutral, or negative way. That's where aspect-based sentimental analysis can help, for example in this product review: "The battery life of this camera is too short", an aspect-based classifier would be able to determine that the sentence expresses a negative opinion about the battery life of the product in question.

Multilingual sentiment analysis

Multilingual sentiment analysis can be difficult. It involves a lot of preprocessing and resources. Most of these resources are available online (e.g. sentiment lexicons), while others need to be created (e.g. translated corpora or noise detection algorithms), but you'll need to know how to code to use them. Alternatively, you could detect the language in texts automatically with a language classifier, then train a custom sentiment analysis model to classify texts in the language of your choice.

ABOUT THE PROBLEM

Various Airline companies exist in this day and age where flying has become much more accessible and cheap than before. There is no shortage of competition, so providing a superior customer experience(CX) to their customers can be a massive difference-maker.

For this reason, online reviews can be an extremely valuable source of information to gain customer insights to improve their CX. From the datasets available online, it is easy to draw a general conclusion about the airlines' relative success. But these results alone fall short if their goal is to improve their services. This perfunctory overview fails to provide actionable insight, into the cornerstone, and end goal, of effective sentiment analysis. If these Airline organizations wanted to unpack the what and why behind their reviews, in order to further improve their services, they would need to analyze each and every negative review at a granular level.

But with sentimental analysis tools, they can plug into an experience of easy-to-understand, helpful, and precise data that can be leveraged for the betterment of each of these organizations on an individual level. Using the relative data can help understand the competition and what the audience /customers like about each of the airlines. If used correctly, this data and the analyzed results can help improve every aspect of the business in a manner that impacts both parties.

ABOUT THE PROBLEM

NEW FEATURES ADDED

Existing applications give a general overview of how the sentiments are distributed on a linear scale. The 'Twitter Sentiment Analysis using R' project tries to integrate many abilities that can help visualize the data better.

- The use of the ggplot library from R makes the representation of the data hassle-free.
- The distribution of negative tweets and their corresponding reasons give a general overview of the state for each department.
- The frequency of positive and negative words in the tweets helps understand how often certain sentiments are reinforced.
- The average text length of Negative tweets helps understand that longer tweets tend to be more negative than positive.

IMPROVEMENT

Tone

Tone can be difficult to interpret verbally, and even more difficult to figure out in the written word. Things get even more complicated when one tries to analyze a massive volume of data that can contain both subjective and objective responses. The basis of any good sentiment analysis software includes the ability to decipher subjective statements from objective ones and then find the right tone in them. A good Tone Detector Algorithm can be used for understanding the tone of the reviews better in order to avoid any sort of inconsistencies in the future.

ABOUT THE PROBLEM

Sarcasm

People use irony and sarcasm in casual conversations and memes on social media. The act of expressing negative sentiment using backhanded compliments can make it difficult for sentiment analysis tools to detect the true context of what the response is actually implying. This can often result in a higher volume of “positive” feedback that is actually negative. For this, the language dataset on which the sentiment analysis model has been trained needs to not only be precise but also massive. In order to correctly identify Sarcasm in such scenarios, the project can integrate a smart model which is well trained over Sarcastic Phrases and sentences.

Emojis

A top-tier sentiment analysis API will be able to detect the context of the language used and everything else involved in creating actual sentiment when a person posts something. To meet sentiment analysis challenges like this, the application needs to employ an emotion analyzer tool that can decode the language in emojis and not club them with special characters like commas, spaces, or full stops.

Idioms

Machine learning programs don’t necessarily understand a figure of speech. For example, an idiom like “not my cup of tea” will boggle the algorithm because it understands things in the literal sense. To overcome this problem a sentiment analysis platform needs to be trained in understanding idioms.

USE / APPLICATION

The overall benefits of sentiment analysis include:

- Sorting Data at Scale
- Real-Time Analysis
- Consistent criteria

Applications of Sentiment Analysis-

1. Social media monitoring
2. Customer support ticket analysis
3. Brand monitoring and reputation management
4. Listen to the voice of the customer (VOC)
5. Listen to the voice of the employee
6. Product analysis
7. Market research and competitive research

Restructuring the project for different Usecases -

The modular nature of the project allows us to run the algorithm on any dataset without any major changes to the code. It can be effectively used for different businesses, different websites using different sorts of inputs. In a way, this algorithm can be applied to almost all businesses having a growth mindset. Starting from social media sites to Educational institutions, almost all areas of life can be impacted using this application. Moreover, the data can be obtained from any platform using Web Mining and Crawling tools.

CODE

Airline Data Sentiment Analysis

Gursheen Kaur Anand(221)

- Preparations
- Sentiment Analysis
- General Info
  - Preliminary visual inspection
  - Findings
- Most frequent terms used and a Wordcloud
- Basic Model
  - Document data
  - Seperate data
  - Randomforest

Preparations

Loading libraries

```
library(dplyr) #Data manipulation
library(forcats) #ggplot frequency
library(ggplot2) #visualizations
library(caTools) #Data wrangling
library("tm") # for text mining
library("SnowballC") # for text stemming
library("wordcloud") # word-cloud generator
library("RColorBrewer") # color palettes
library(randomForest) #randomforest
library(sentimentr)
```

Loading data

```
df = read.csv("G:\\Gursheen\\R_NLP\\Tweets1.csv")
```

CODE

Sentiment Analysis

```
df$ave_sentiment=0
df$sentiment=0
df$positivereason="Unspecified"
df$negativereason="Unspecified"
for(i in 1:15){
  sentiment=sentiment_by(df[i,4])
  df[i,5]=sentiment$ave_sentiment
  if(sentiment$ave_sentiment<0){
    df[i,6]='negative'
    df[i,9]=df[i,7]
  }
  else if(sentiment$ave_sentiment==0){
    df[i,6]='neutral'
  }
  else if(sentiment$ave_sentiment>0){
    df[i,6]='positive'
    df[i,8]=df[i,7]
  }
}
```

General Info

```
df
```

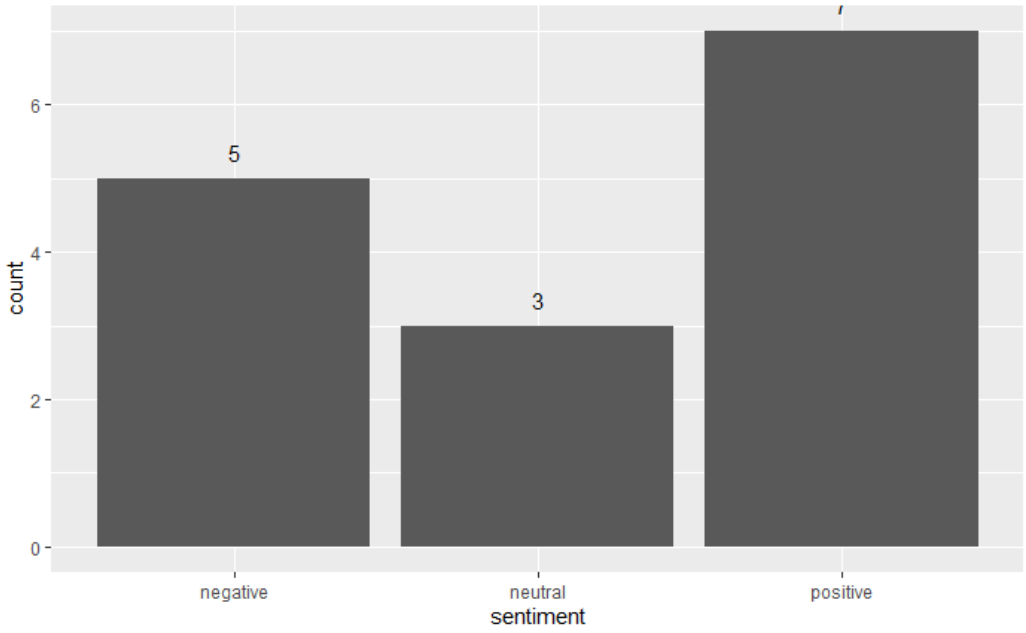
```
'data.frame':   15 obs. of  10 variables:
 $ tweet_id      : num  5.70e+17 5.80e+17 5.90e+17 6.70 6.87 ...
 $ airline       : chr  "Virgin America" "Virgin America" "Virgin America" "United" ...
 $ name          : chr  "Rebecca Morales" "Jessica Wong" "Laura Dominguez" "Aaron Horne" ...
 $ text          : chr  "@VirginAmericaStaff at the counter mentioned that if was usual for the
airline to cancel flights and that the c"| __truncated__ "@VirginAmerica Good experience."
"@VirginAmerica So sad that they no longer operating, flew one the last flights on THE last day
operating when t"| __truncated__ "@United Small plane but was rather empty so space enough,
smooth flight on time with early arrival. Good info o"| __truncated__ ...
 $ ave_sentiment : num  -0.245 0.433 0.334 0.199 0.785 ...
 $ sentiment     : chr  "negative" "positive" "positive" "positive" ...
 $ reason        : chr  "Cancelled flight" "Customer Service" "Customer Service" "All Services"
...
 $ positivereason: chr  "Unspecified" "Customer Service" "Customer Service" "All Services" ...
 $ negativereason: chr  "Cancelled flight" "Unspecified" "Unspecified" "Unspecified" ...
 $ length        : int  161 31 173 215 178 203 26 178 797 294 ...
```

CODE

Preliminary visual inspection

Total distribution of tweets with sentiment

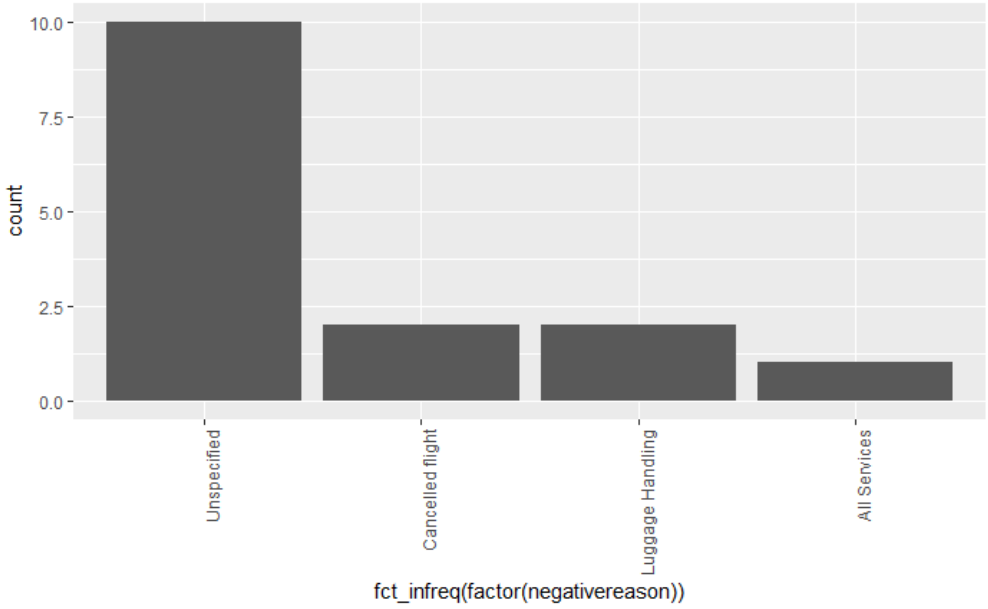
```
#Total tweets distribution as negative,positive,neutral
ggplot(df, aes(x = sentiment))+geom_bar(stat = "count")+
geom_text(stat='count', aes(label=..count..), vjust=-1)
```



Distribution of Negative Tweets and their Reasons

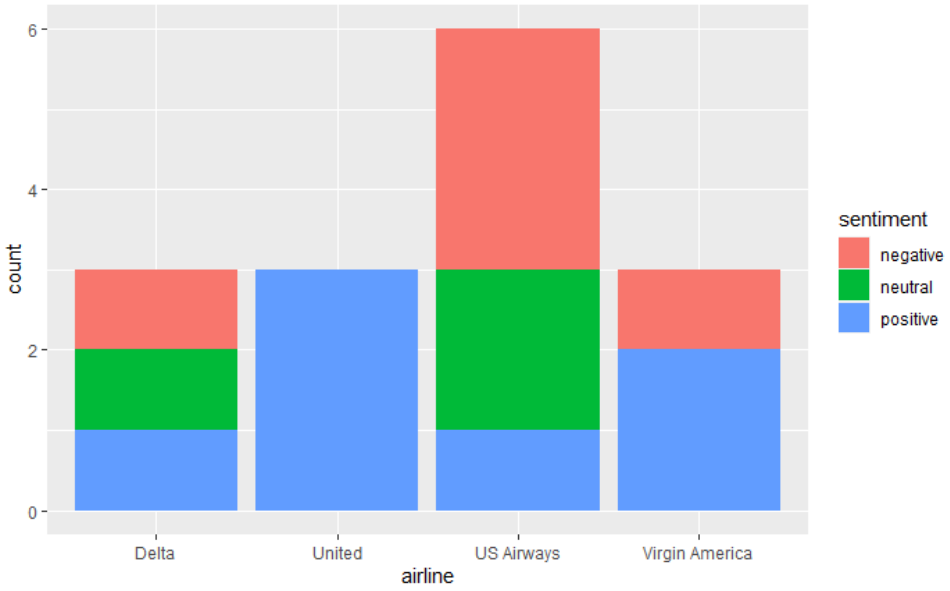
```
#fct_infreq is from package forcats (for frequency
distribution)
ggplot(df, aes(x =
fct_infreq(factor(negativereason))))+geom_bar(stat =
"count")+theme(axis.text.x = element_text(angle = 90, hjust =
1))
```

CODE



Distribution of Sentiments for different Airlines.

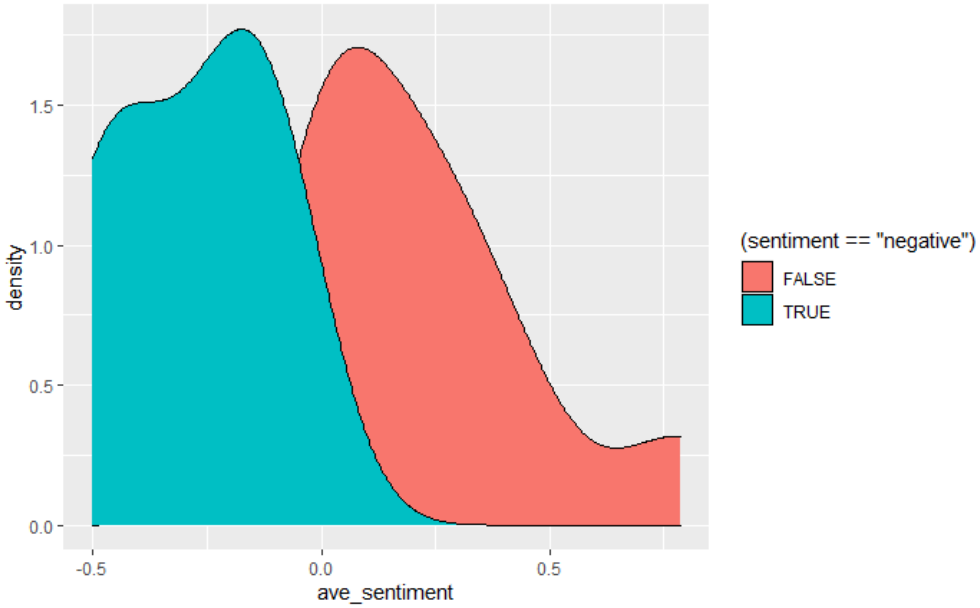
```
ggplot(df, aes(x = airline,fill = sentiment ))+geom_bar(stat
= "count")
```



CODE

Only Negative tweets

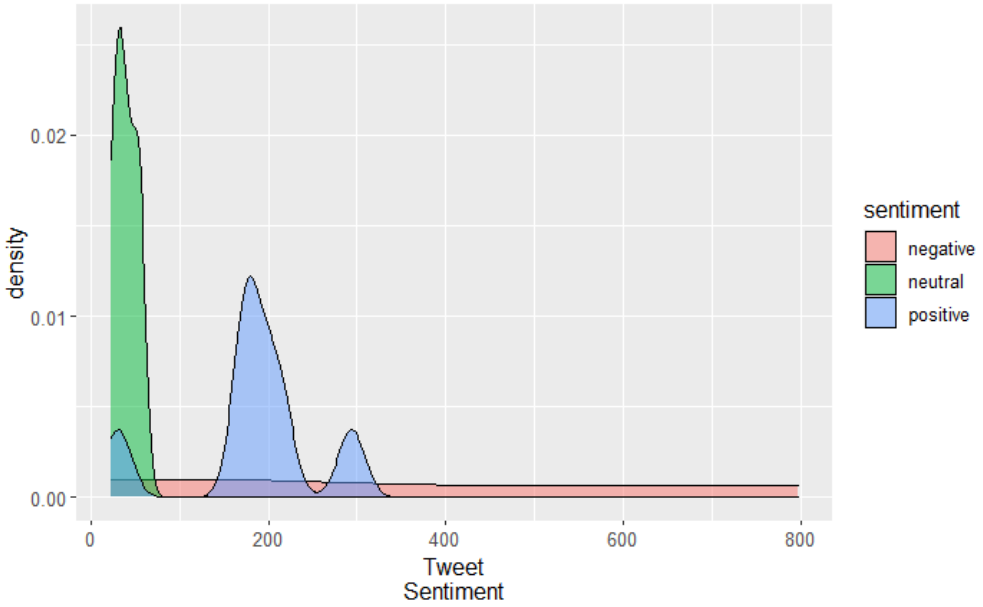
```
ggplot(df,aes(x=ave_sentiment,fill=(sentiment=="negative"
)))+geom_density()
```



Distribution of Text length for various Sentiments

```
#creating a variable with text length of each tweet
df$length = nchar(as.character(df$text))
ggplot(df, aes(x = length, fill = sentiment))+
geom_density(alpha=0.5)+scale_x_continuous(name   =
'Tweet\nSentiment') +labs(x='Tweet Length') +theme(text =
element_text(size=12))
```

CODE



Findings

- On Total, there are 5 negative tweets compared to 3 neutral 7 positive tweets.
- Canceled Flight is the leading reason for negative tweets.
- United Airlines has the most no. of positive tweets.
- US Airways has the highest number of tweets.
- Neutral Tweets are generally shorter compared to other tweets.

CODE

Most frequent terms used and a Wordcloud

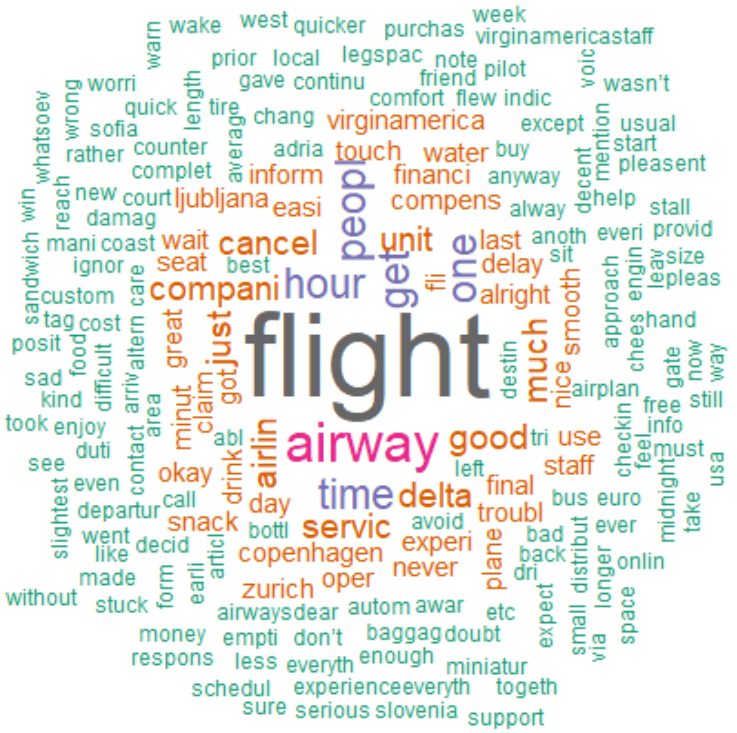
```
# Load the data as a corpus
docs <- VCorpus(VectorSource(df$text))
#To replace special characters
toSpace <- content_transformer(function (x , pattern )
gsub(pattern, " ", x))
docs <- tm_map(docs, toSpace, "/")
docs <- tm_map(docs, toSpace, "@")
docs <- tm_map(docs, toSpace, "\\|")
#convert to lower case
docs <- tm_map(docs,content_transformer(tolower))
# Remove numbers
docs <- tm_map(docs, removeNumbers)
# Remove english common stopwords
docs <- tm_map(docs, removeWords, stopwords("english"))
# Remove punctuations
docs <- tm_map(docs, removePunctuation)
# Eliminate extra white spaces
docs <- tm_map(docs, stripWhitespace)
# Text stemming
docs <- tm_map(docs, stemDocument)

#Text to Matrix
tdm <- TermDocumentMatrix(docs)
m <- as.matrix(tdm)
v <- sort(rowSums(m),decreasing=TRUE)
d <- data.frame(word = names(v),freq=v)
head(d, 10)
```

CODE

	word <chr>	freq <dbl>
flight	flight	15
airway	airway	6
get	get	4
hour	hour	4
one	one	4
peopl	peopl	4
time	time	4
airlin	airlin	3
cancel	cancel	3
compani	compani	3

```
#word Cloud
set.seed(1234)
wordcloud(words = d$word, freq = d$freq, min.freq = 1,
          max.words=200, random.order=FALSE, rot.per=0.35,
          colors=brewer.pal(8, "Dark2"))
```



CODE

Basic Model

Document data

```
dtm = DocumentTermMatrix(docs)
dtm = removeSparseTerms(dtm,sparse = 0.99)
model_data = as.data.frame(as.matrix(dtm))
model_data$sentiment = df$sentiment
```

Separate data

```
model_data$sentiment <- as.factor(model_data$sentiment)
set.seed(123)
split = sample.split(model_data$sentiment,SplitRatio = 0.9)
train = subset(model_data,split = TRUE)
test = subset(model_data,split = FALSE)
```

Randomforest

```
predict_rf = randomForest(x = train[,-159],y =
train$sentiment,ntree = 10)
y_pred = predict(predict_rf,newdata = test[,-159])
y_pred
#confusion matrix
cm = table(test[,159],y_pred)
cm
#Accuracy
sum(diag(cm))/sum(cm)
```

```
       1        2        3        4        5        6        7        8
negative positive positive positive positive positive  neutral positive
       9       10       11       12       13       14       15
negative positive negative negative  neutral  neutral negative
Levels: negative neutral positive
   y_pred
    negative neutral positive
  0        5       3        6
  1        0       0        1
[1] 0.3333333
```

EXAMPLES

Basic examples of sentiment analysis data

- Netflix has the best selection of films
- Hulu has a great UI
- I dislike the new crime series
- I hate waiting for the next series to come out

More challenging examples of sentiment analysis

- I do not dislike horror movies. (a phrase with negation)
- Disliking horror movies is not uncommon. (negation, inverted word order)
- Sometimes I really hate the show. (adverbial modifies the sentiment)
- I love having to wait two months for the next series to come out! (sarcasm)

Application-Specific Examples -

- Got such a great deal! - Positive
- Good staff, Comfort was made a priority, overall enjoyed my experience. - Positive
- Got such a bad deal! - Negative
- The flight got delayed by 2 hours. Tiresome. - Negative
- No offers whatsoever, the experience was okay. - Neutral
- Not extraordinary.An alright experience. - Neutral

SOFTWARE & LIBRARIES

Softwares -

- R v4.1.3
- RStudio IDE

Libraries -

- dplyr - For Data manipulation
- sentimentr - For Sentiment scores
- forcats - For ggplot frequency
- ggplot2 - For visualizations and plots
- caTool - For Data wrangling
- tm - For text mining
- SnowballC - For text stemming
- Wordcloud - For word-cloud generator
- RColorBrewer - For color palettes
- randomForest - For randomforest

RESOURCES

Understanding the concept -

https://towardsdatascience.com/sentiment-analysis-concept-analysis-and-applications-6c94d6f58c17

https://monkeylearn.com/sentiment-analysis/

R version 4.1.3 download -

https://cran.r-project.org/bin/windows/base/

RStudio IDE download -

https://www.rstudio.com/products/rstudio/download/

Basic R Commands -

https://www.tutorialspoint.com/r/index.htm

Introduction to Sentimentr Library -

https://towardsdatascience.com/doing-your-first-sentiment-analysis-in-r-with-sentimentr-167855445132

Introduction to ggplot2 Library -

http://r-statistics.co/Complete-Ggplot2-Tutorial-Part1-With-R-Code.html