Text Mining Classification Case Study

Niranjan

1 January 2019

Importing the variable and examining it

```
yelp <- fread("yelp.csv")
str(yelp)
```

```
## Classes 'data.table' and 'data.frame':   10000 obs. of  10 variables:
##  $ business_id: chr  "9yKzy9PApeiPPOUJEtnvkg" "ZRJwVLyzEJq1VAihDhYiow" "6oRAC4uyJCsJl1X0WZ
pVSA" "_1QQZuf4zZOyFCvXc0o6Vg" ...
##  $ date       : chr  "2011-01-26" "2011-07-27" "2012-06-14" "2010-05-27" ...
##  $ review_id  : chr  "fWKvX83p0-ka4JS3dc6E5A" "IjZ33sJrzXqU-0X6U8NwyA" "IESLBzqUCLdSzSqm0e
CSxQ" "G-WvGaISbqqaMHlNnByodA" ...
##  $ stars      : int  5 5 4 5 5 4 5 4 4 5 ...
##  $ text       : chr  "My wife took me here on my birthday for breakfast and it was excelle
nt.  The weather was perfect which made sit"| __truncated__ "I have no idea why some people g
ive bad reviews about this place. It goes to show you, you can please everyone."| __truncated
__ "love the gyro plate. Rice is so good and I also dig their candy selection :)" "Rosie, Dak
ota, and I LOVE Chaparral Dog Park!!! It's very convenient and surrounded by a lot of paths,
a desert"| __truncated__ ...
##  $ type       : chr  "review" "review" "review" "review" ...
##  $ user_id    : chr  "rLtl8ZkDX5vH5nAx9C3q5Q" "0a2KyEL0d3Yb1V6aivbIuQ" "0hT2KtfLiobPvh6cDC
8JQg" "uZetl9T0NcROGOyFfughhg" ...
##  $ cool       : int  2 0 0 1 0 4 7 0 0 0 ...
##  $ useful     : int  5 0 1 2 0 3 7 1 0 1 ...
##  $ funny      : int  0 0 0 0 0 1 4 0 0 0 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

```
head(yelp)
```

```
##               business_id       date              review_id stars
## 1: 9yKzy9PApeiPPOUJEtnvkg 2011-01-26 fWKvX83p0-ka4JS3dc6E5A     5
## 2: ZRJwVLyzEJq1VAihDhYiow 2011-07-27 IjZ33sJrzXqU-0X6U8NwyA     5
## 3: 6oRAC4uyJCsJl1X0WZpVSA 2012-06-14 IESLBzqUCLdSzSqm0eCSxQ     4
## 4: _1QQZuf4zZOyFCvXc0o6Vg 2010-05-27 G-WvGaISbqqaMHlNnByodA     5
## 5: 6ozycU1RpktNG2-1BroVtw 2012-01-05 1uJFq2r5QfJG_6ExMRCaGw     5
## 6: -yxfBYGB6SEqszmxJxd97A 2007-12-13 m2CKSsepBCoRYWxiRUsxAg     4
##




























                                                                  text
## 1:
















  My wife took me here on my birthday for breakfast and it was excellent.  The weather was pe
rfect which made sitting outside overlooking their grounds an absolute pleasure.  Our waitres
s was excellent and our food arrived quickly on the semi-busy Saturday morning.  It looked li
ke the place fills up pretty quickly so the earlier you get here the better.\n\nDo yourself a
favor and get their Bloody Mary.  It was phenomenal and simply the best I've ever had.  I'm p
retty sure they only use ingredients from their garden and blend them fresh when you order i
t.  It was amazing.\n\nWhile EVERYTHING on the menu looks excellent, I had the white truffle
scrambled eggs vegetable skillet and it was tasty and delicious.  It came with 2 pieces of th
eir griddled bread with was amazing and it absolutely made the meal complete.  It was the bes
t ""toast"" I've ever had.\n\nAnyway, I can't wait to go back!
## 2:
```

I have no idea why some people give bad reviews about this place. It goes to show you, you can please everyone. They are probably griping about something that their own fault...there are many people like that.\n\nIn any case, my friend and I arrived at about 5:50 PM this past Sunday. It was pretty crowded, more than I thought for a Sunday evening and thought we would have to wait forever to get a seat but they said we'll be seated when the girl comes back from seating someone else. We were seated at 5:52 and the waiter came and got our drink orders. Everyone was very pleasant from the host that seated us to the waiter to the server. The prices were very good as well. We placed our orders once we decided what we wanted at 6:02. We shared the baked spaghetti calzone and the small ""Here's The Beef"" pizza so we can both try them. The calzone was huge and we got the smallest one (personal) and got the small 11"" pizza. Both were awesome! My friend liked the pizza better and I liked the calzone better. The calzone does have a sweetish sauce but that's how I like my sauce!\n\nWe had to box part of the pizza to take it home and we were out the door by 6:42. So, everything was great and not like these bad reviewers. That goes to show you that you have to try these things yourself because all these bad reviewers have some serious issues.

3:

love the gyro plate. Rice is so good and I also dig their candy selection :)

4:

```
          Rosie, Dakota, and I LOVE Chaparral Dog Park!!! It's very convenient and surrounde
d by a lot of paths, a desert xeriscape, baseball fields, ballparks, and a lake with ducks.\n
\nThe Scottsdale Park and Rec Dept. does a wonderful job of keeping the park clean and shade
d.  You can find trash cans and poopy-pick up mitts located all over the park and paths.\n\nT
he fenced in area is huge to let the dogs run, play, and sniff!
## 5:

                                                     General Manager Scott Petello is a good
egg!!! Not to go into detail, but let me assure you if you have any issues (albeit rare) spea
k with Scott and treat the guy with some respect as you state your case and I'd be surprised
if you don't walk out totally satisfied as I just did. Like I always say..... ""Mistakes are
inevitable, it's how we recover from them that is important""!!!\n\nThanks to Scott and his a
wesome staff. You've got a customer for life!! .......... :^)
## 6: Quiessence is, simply put, beautiful.  Full windows and earthy wooden walls give a feel
ing of warmth inside this restaurant perched in the middle of a farm.  The restaurant seemed
fairly full even on a Tuesday evening; we had secured reservations just a couple days befor
e.\n\nMy friend and I had sampled sandwiches at the Farm Kitchen earlier that week, and were
impressed enough to want to eat at the restaurant.  The crisp, fresh veggies didn't disappoin
t: we ordered the salad with orange and grapefruit slices and the crudites to start.  Both we
re very good; I didn't even know how much I liked raw radishes and turnips until I tried them
with their pesto and aioli sauces.\n\nFor entrees, I ordered the lamb and my friend ordered t
he pork shoulder.  Service started out very good, but trailed off quickly.  Waiting for our f
ood took a very long time (a couple seated after us received and finished their entrees befor
e we received our's), and no one bothered to explain the situation until the maitre'd apologi
zed almost 45 minutes later.  Apparently the chef was unhappy with the sauce on my entree, so
he started anew.  This isn't really a problem, but they should have communicated this to us e
arlier.  For our troubles, they comped me the glass of wine I ordered, but they forgot to bri
ng out with my entree  as I had requested.  Also, they didn't offer us bread, but I will echo
the lady who whispered this to us on her way out: ask for the bread.  We received warm foccac
ia, apple walnut, and pomegranate slices of wonder with honey and butter.  YUM.\n\nThe entree
s were both solid, but didn't quite live up to the innovation and freshness of the vegetable
s.  My lamb's sauce was delicious, but the meat was tough.  Maybe the vegetarian entrees are
the way to go?  But our dessert, the gingerbread pear cake, was yet another winner.\n\nIf the
entrees were tad more inspired, or the service weren't so spotty, this place definitely would
have warranted five stars.  If I return, I'd like to try the 75$ tasting menu.  Our bill came
out to about 100$ for two people, including tip, no drinks.
##      type                user_id cool useful funny
## 1: review rLtl8ZkDX5vH5nAx9C3q5Q    2      5     0
## 2: review 0a2KyEL0d3Yb1V6aivbIuQ    0      0     0
## 3: review 0hT2KtfLiobPvh6cDC8JQg    0      1     0
## 4: review uZetl9T0NcROGOyFfughhg    1      2     0
```

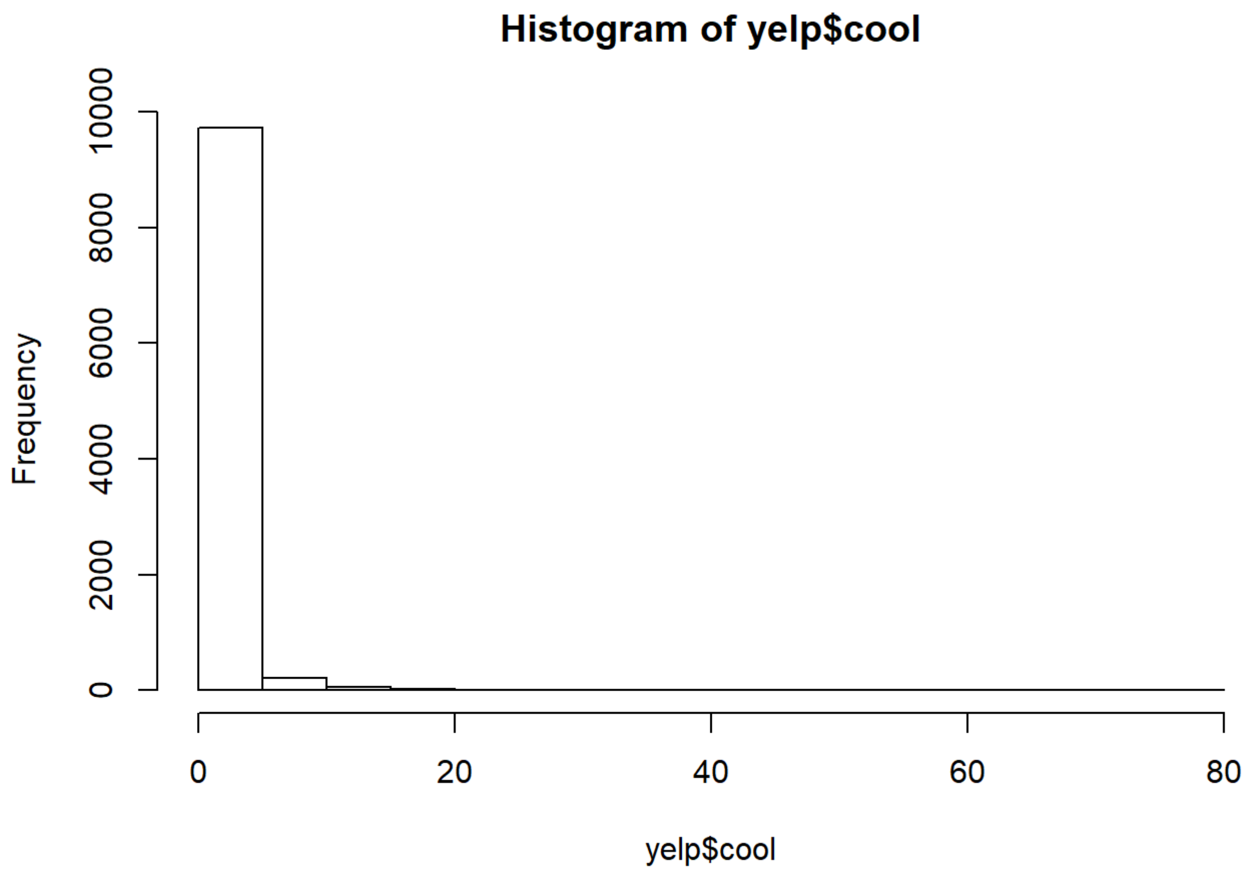
```
## 5: review vYmM4KTsC8ZfQBg-j5MWkw    0      0     0
## 6: review sqYN3lNgvPbPCTRsMFu27g    4      3     1
```

of all the variables, it is certain that the date and type variable is not much relevance for the analysis, so it is removed

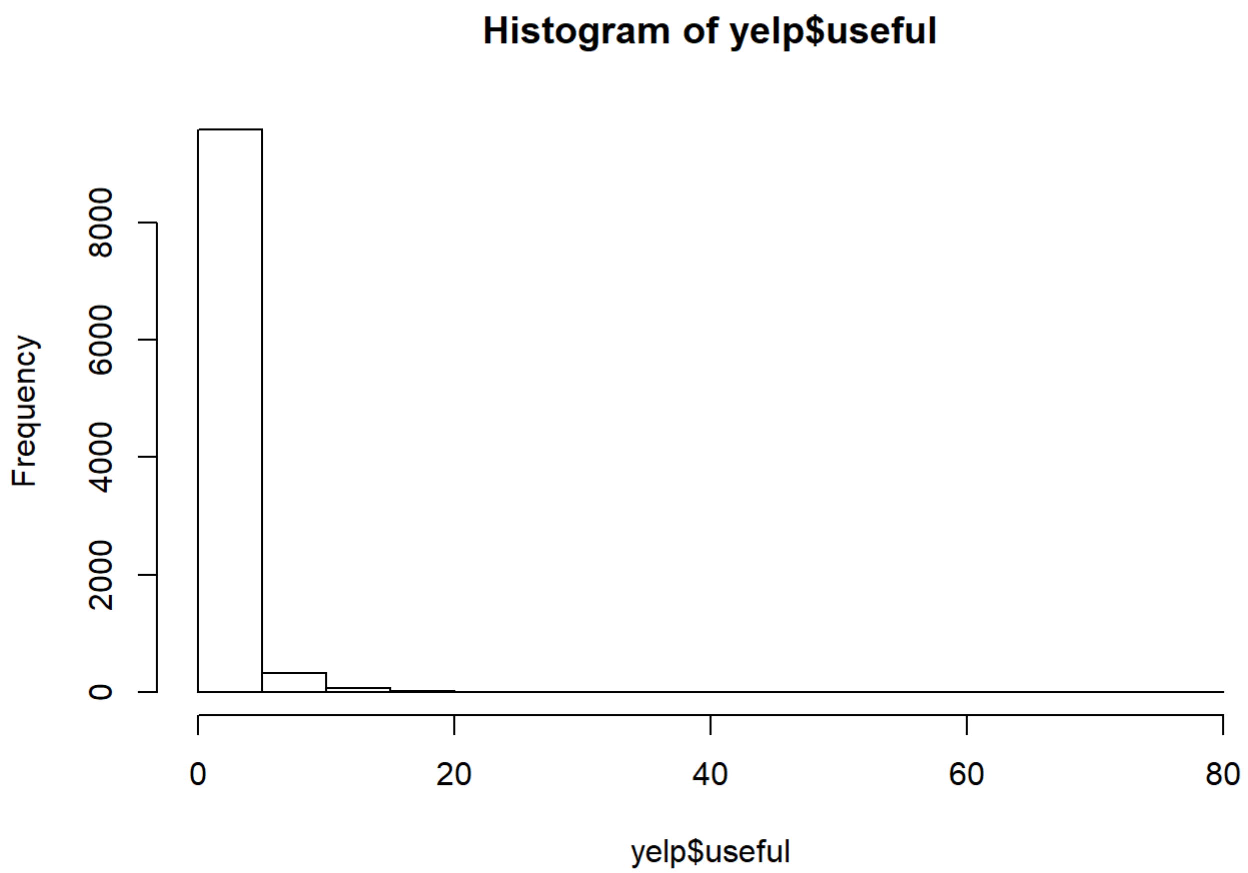
```
yelp$date <- NULL
yelp$type <- NULL
```

First the existing cool, useful and funny variables are examined

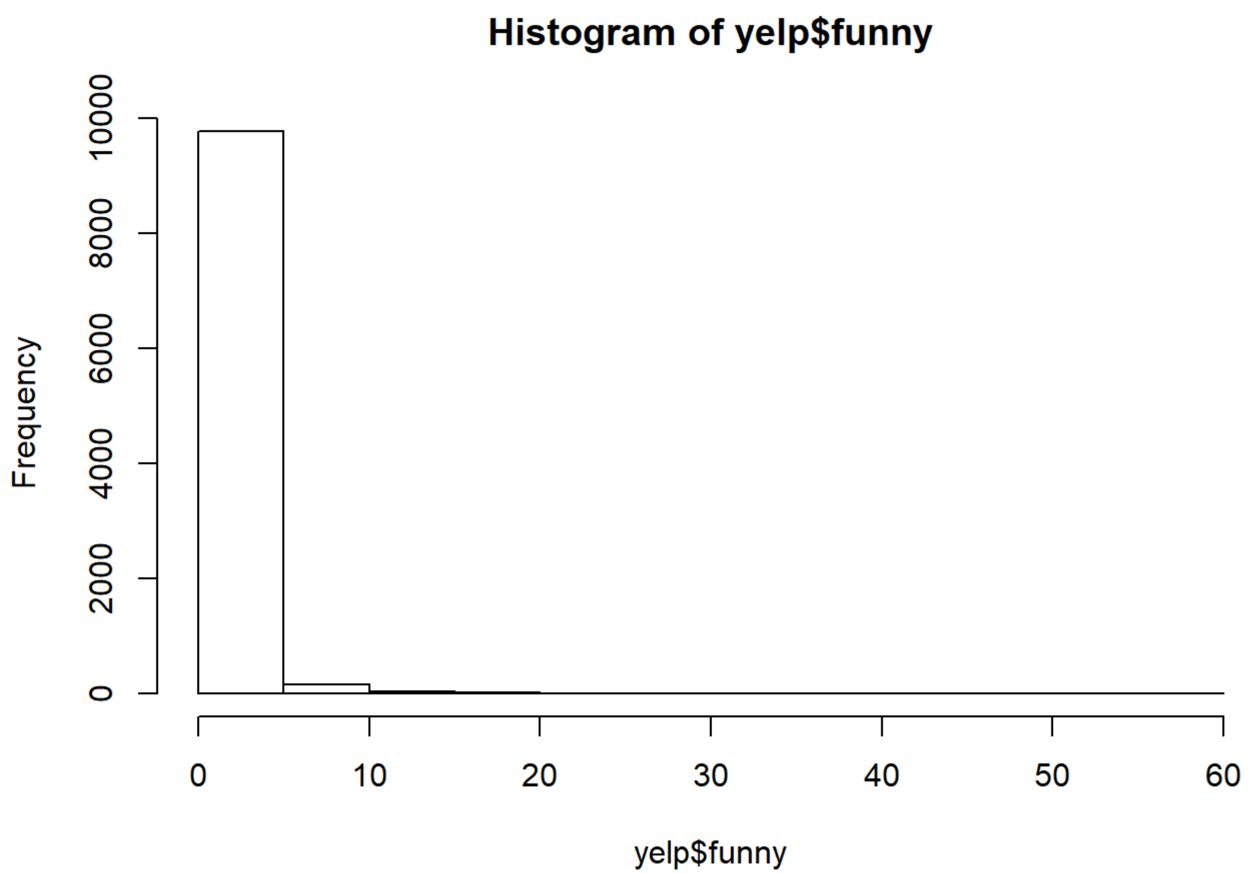
```
hist(yelp$cool)
```




```
hist(yelp$useful)
```




```
hist(yelp$funny)
```



```r
data.table(prop.table(table(yelp$cool))*100)
```

```
##     V1     N
##  1:  0 62.90
##  2:  1 19.55
##  3:  2  7.49
##  4:  3  3.96
##  5:  4  2.09
##  6:  5  1.19
##  7:  6  0.88
##  8:  7  0.41
##  9:  8  0.31
## 10:  9  0.15
## 11: 10  0.30
## 12: 11  0.17
## 13: 12  0.09
## 14: 13  0.14
## 15: 14  0.10
## 16: 15  0.05
## 17: 16  0.06
## 18: 17  0.05
## 19: 18  0.01
## 20: 19  0.01
## 21: 20  0.01
## 22: 21  0.01
## 23: 22  0.01
## 24: 23  0.01
## 25: 27  0.01
## 26: 28  0.01
## 27: 32  0.01
## 28: 38  0.01
## 29: 77  0.01
##     V1     N
```

```r
data.table(prop.table(table(yelp$useful))*100)
```

```
##     V1     N
##  1:  0 41.30
##  2:  1 28.48
##  3:  2 13.23
##  4:  3  7.11
##  5:  4  3.35
##  6:  5  2.22
##  7:  6  1.14
##  8:  7  0.91
##  9:  8  0.52
## 10:  9  0.38
## 11: 10  0.29
## 12: 11  0.19
## 13: 12  0.20
## 14: 13  0.12
## 15: 14  0.08
## 16: 15  0.17
## 17: 16  0.06
## 18: 17  0.05
## 19: 18  0.05
## 20: 19  0.06
## 21: 20  0.02
## 22: 23  0.01
## 23: 24  0.01
## 24: 28  0.01
## 25: 30  0.01
## 26: 31  0.01
## 27: 38  0.01
## 28: 76  0.01
##     V1     N
```

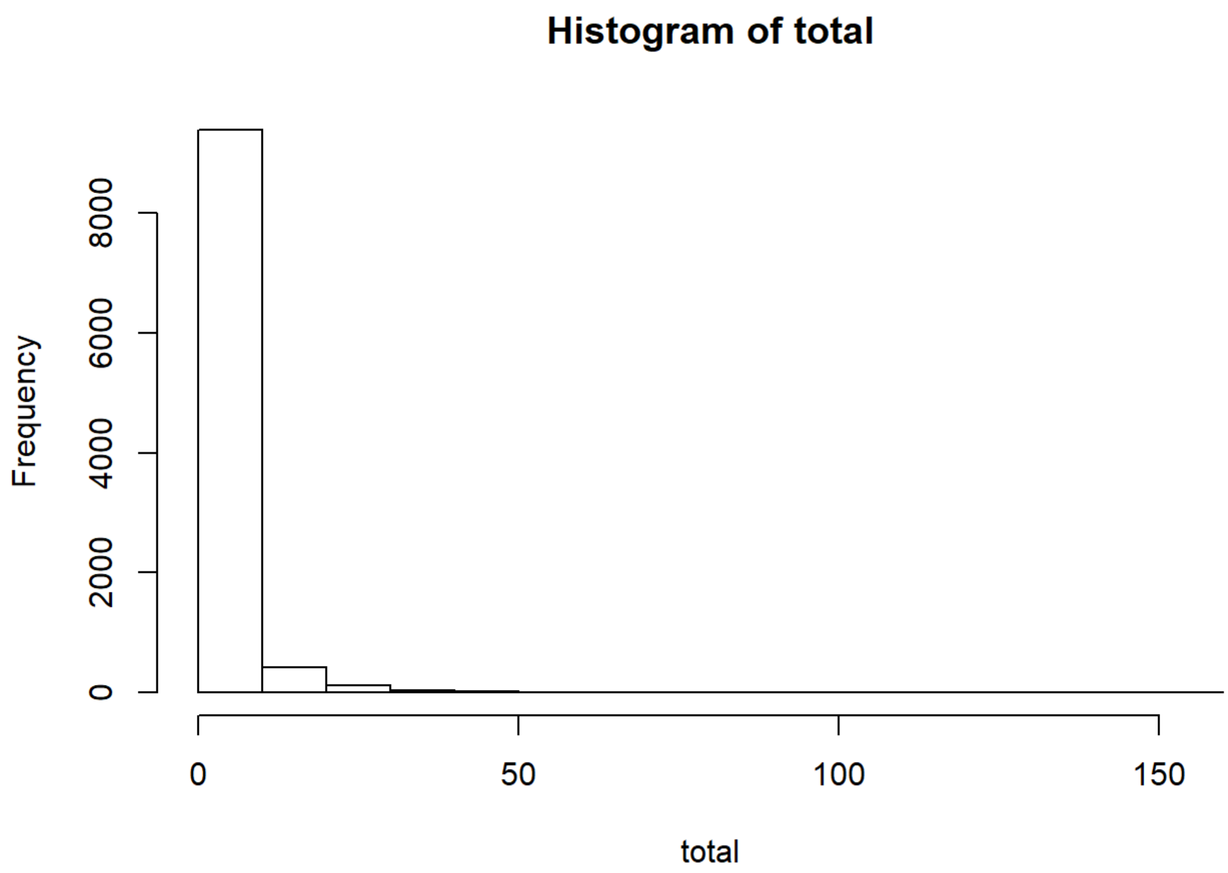
```
data.table(prop.table(table(yelp$funny))*100)
```

```
##     V1     N
##  1:  0 70.13
##  2:  1 16.32
##  3:  2  6.03
##  4:  3  2.71
##  5:  4  1.61
##  6:  5  0.90
##  7:  6  0.68
##  8:  7  0.34
##  9:  8  0.25
## 10:  9  0.21
## 11: 10  0.18
## 12: 11  0.14
## 13: 12  0.11
## 14: 13  0.03
## 15: 14  0.03
## 16: 15  0.05
## 17: 16  0.05
## 18: 17  0.04
## 19: 18  0.02
## 20: 19  0.02
## 21: 20  0.05
## 22: 21  0.01
## 23: 22  0.02
## 24: 23  0.01
## 25: 24  0.02
## 26: 27  0.01
## 27: 30  0.01
## 28: 39  0.01
## 29: 57  0.01
##     V1     N
```

The histogram and the proportion tables shows that there are a lot of zero values in them

The same at an overall level for all cool,useful and funny is examined too

```
total <- yelp$cool+yelp$useful+yelp$funny
hist(total)
```




```
data.table(prop.table(table(total))*100)
```

```
##     total     N
##  1:     0 36.12
##  2:     1 19.44
##  3:     2 10.82
##  4:     3  9.24
##  5:     4  5.63
##  6:     5  3.87
##  7:     6  2.81
##  8:     7  2.16
##  9:     8  1.54
## 10:     9  1.14
## 11:    10  1.10
## 12:    11  0.87
## 13:    12  0.77
## 14:    13  0.48
## 15:    14  0.49
## 16:    15  0.47
## 17:    16  0.27
## 18:    17  0.32
## 19:    18  0.23
## 20:    19  0.18
## 21:    20  0.12
## 22:    21  0.17
## 23:    22  0.12
## 24:    23  0.22
## 25:    24  0.17
## 26:    25  0.11
## 27:    26  0.09
## 28:    27  0.06
## 29:    28  0.10
## 30:    29  0.05
## 31:    30  0.07
## 32:    31  0.05
## 33:    32  0.03
## 34:    33  0.06
## 35:    34  0.04
## 36:    35  0.04
## 37:    36  0.03
## 38:    37  0.06
## 39:    38  0.04
## 40:    39  0.03
## 41:    40  0.01
## 42:    41  0.03
## 43:    42  0.03
## 44:    43  0.01
## 45:    44  0.03
## 46:    45  0.03
## 47:    46  0.01
## 48:    47  0.03
## 49:    48  0.02
## 50:    49  0.01
## 51:    50  0.02
## 52:    51  0.01
## 53:    52  0.01
## 54:    54  0.01
## 55:    55  0.01
## 56:    57  0.01
```

```
## 57:    58  0.02
## 58:    59  0.02
## 59:    67  0.01
## 60:    72  0.01
## 61:    82  0.02
## 62:    95  0.01
## 63:   133  0.01
## 64:   153  0.01
##     total     N
```

36% of the total reviews have zeros as as cool, funny and useful

Relationship of these variables with the stars needs to be calculated

```
yelp$stars <- as.character(yelp$stars)

summary(aov(yelp$stars~yelp$cool))
```

```
##                Df Sum Sq Mean Sq F value   Pr(>F)    
## yelp$cool      1     41   40.74   27.69 1.45e-07 ***
## Residuals   9998  14711    1.47                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(aov(yelp$stars~yelp$useful))
```

```
##                Df Sum Sq Mean Sq F value Pr(>F)  
## yelp$useful    1      8   8.132   5.515 0.0189 *
## Residuals   9998  14744   1.475                 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(aov(yelp$stars~yelp$funny))
```

```
##                Df Sum Sq Mean Sq F value   Pr(>F)    
## yelp$funny     1     55   55.44   37.72 8.48e-10 ***
## Residuals   9998  14696    1.47                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

ANOVA shows that the cool, useful and funny variables are significantly different between the star ratings.Which means that there is a relationship between them and star ratings.

Inorder to analyze the reviews, some data engineering steps needs to be done

```
r_words <- yelp[,c(1:4)]
r_words <- unnest_tokens(r_words,word,text)
r_words <- r_words[!word %in% stop_words$word]
r_words <- r_words[str_detect(r_words$word,"^[a-z']+$")]
```

Now that the cleaning is done, the sentiment value for each of the words is ascertained using the AFINN Lexicon and the sentiments from nrc is taken

```
sentiments <- as.data.table(sentiments)
AFINN <- sentiments[lexicon == "AFINN",c(1,4),]
NRC <- sentiments[lexicon == "nrc",c(1,2),]

r_senti <- data.table(inner_join(r_words,AFINN,by="word"))
r_senti_1 <- r_senti[,.(sentiment = mean(score)),by=.(review_id,stars)]
r_senti_1 <- r_senti_1[,c(1,3),]

r_senti <- data.table(inner_join(r_senti,NRC,by="word"))
r_senti <- dummy_cols(r_senti,"sentiment")
r_senti_2 <- r_senti[,.(sentiment_joy=sum(sentiment_joy),
                        sentiment_positive=sum(sentiment_positive),
                        sentiment_trust=sum(sentiment_trust),
                        sentiment_anticipation=sum(sentiment_anticipation),
                        sentiment_anger= sum(sentiment_anger),
                        sentiment_disgust=sum(sentiment_disgust),
                        sentiment_fear=sum(sentiment_fear),
                        sentiment_negative=sum(sentiment_negative),
                        sentiment_sadness=sum(sentiment_sadness)),by=.(review_id,stars)]
r_senti_2$stars <- NULL
```

The average sentiment score and the sentiment is added up to the main yelp data

```
yelp <- data.table(inner_join(yelp,r_senti_1,by="review_id"))
yelp <- data.table(inner_join(yelp,r_senti_2,by="review_id"))
```

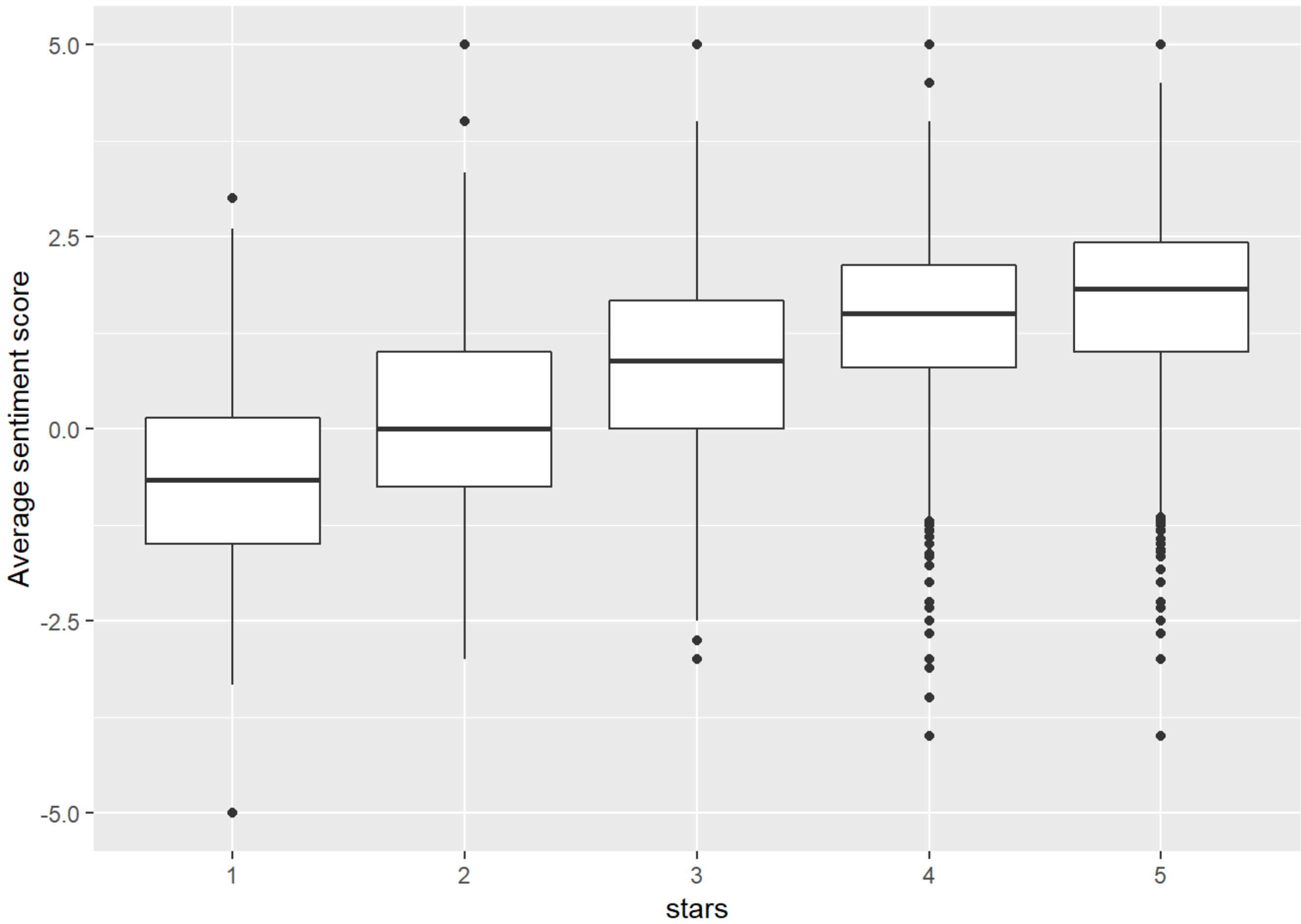
The relationship between average sentiment score and the sentiment words with the Star ratings is examined

Sentiment scores

```
summary(aov(yelp$stars~yelp$sentiment))
```

```
##                  Df Sum Sq Mean Sq F value Pr(>F)
## yelp$sentiment    1   3131  3130.8    2810 <2e-16 ***
## Residuals      8805   9809     1.1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
ggplot(yelp, aes(stars, sentiment, group = stars)) +
  geom_boxplot() +
  ylab("Average sentiment score")
```



The ANOVA establishes that there is a relationship between the average sentiment score and the star rating. The plot confirms this relationship, but there does seem to be a lot of outliers with lower sentiment scores for 4&5 stars

Sentiment words

```
summary(aov(yelp$stars~yelp$sentiment_joy))
```

```
##                        Df Sum Sq Mean Sq F value Pr(>F)    
## yelp$sentiment_joy    1    384   384.1   269.3 <2e-16 ***
## Residuals          8805  12556     1.4                   
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(aov(yelp$stars~yelp$sentiment_positive))
```

```
##                             Df Sum Sq Mean Sq F value Pr(>F)    
## yelp$sentiment_positive    1    322   321.6   224.4 <2e-16 ***
## Residuals               8805  12619     1.4                   
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(aov(yelp$stars~yelp$sentiment_trust))
```

```
##                         Df Sum Sq Mean Sq F value Pr(>F)    
## yelp$sentiment_trust    1    172  171.71   118.4 <2e-16 ***
## Residuals            8805  12768    1.45                   
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(aov(yelp$stars~yelp$sentiment_anticipation))
```

```
##                                Df Sum Sq Mean Sq F value   Pr(>F)    
## yelp$sentiment_anticipation    1     30  29.764    20.3 6.71e-06 ***
## Residuals                   8805  12910   1.466                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(aov(yelp$stars~yelp$sentiment_anger))
```

```
##                         Df Sum Sq Mean Sq F value Pr(>F)    
## yelp$sentiment_anger    1    935   935.4   686.1 <2e-16 ***
## Residuals            8805  12005     1.4                   
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(aov(yelp$stars~yelp$sentiment_disgust))
```

```
##                           Df Sum Sq Mean Sq F value Pr(>F)    
## yelp$sentiment_disgust    1   1394  1393.6    1063 <2e-16 ***
## Residuals              8805  11546     1.3                   
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(aov(yelp$stars~yelp$sentiment_fear))
```

```
##                        Df Sum Sq Mean Sq F value Pr(>F)    
## yelp$sentiment_fear    1    844   844.1   614.4 <2e-16 ***
## Residuals           8805  12096     1.4                   
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

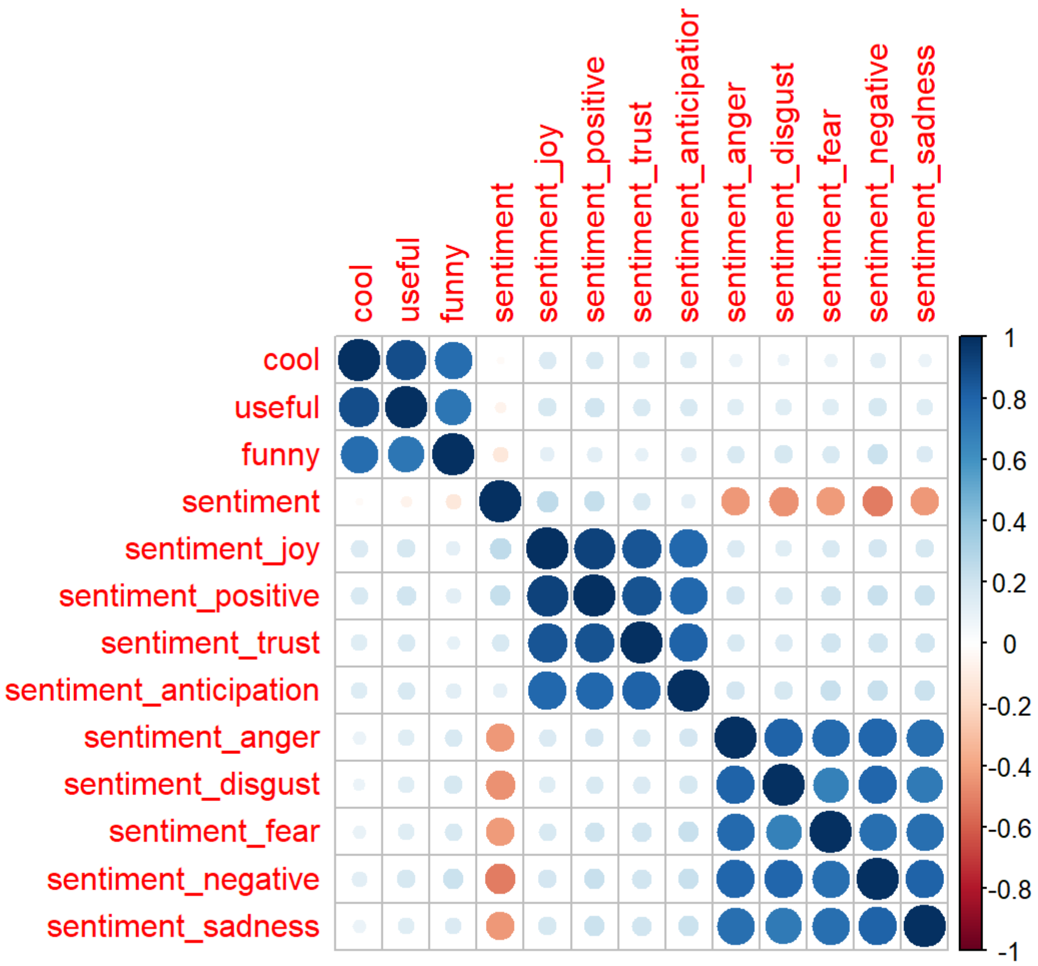
```
summary(aov(yelp$stars~yelp$sentiment_negative))
```

```
##                            Df Sum Sq Mean Sq F value Pr(>F)    
## yelp$sentiment_negative    1   1448  1447.8    1109 <2e-16 ***
## Residuals               8805  11492     1.3                   
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(aov(yelp$stars~yelp$sentiment_sadness))
```

```
##                           Df Sum Sq Mean Sq F value Pr(>F)    
## yelp$sentiment_sadness     1   1007  1006.9     743 <2e-16 ***
## Residuals               8805  11933     1.4                   
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
corrplot::corrplot(cor(select_if(yelp,is.numeric)))
```



All the sentiment words are related to ratings, but there is a strong amount of inter correlation too, that are grouped togeather as mainly positive and negative emotions, thus, only those are kept and other word variables are dropped

```
yelp <- yelp[,-c(10,12:16,18)]
```

Building the model

For this model, the Random Forest model is used

Removing the variables that will overfit the data

```
FA <- yelp[,-c(1,2,4,5)]
FA$stars <- as.factor(FA$stars)
```

Inorder to preserve the proportion of contribution of all groups a stratified random sampling is used

```
library(rsample)

table(FA$stars)
```

```
##
##    1    2    3    4    5
##  658  809 1278 3131 2931
```

```
set.seed(123)
split <- initial_split(FA, prop = .7, strata = "stars")
train <- training(split)
test <- testing(split)
```

Setting parameters and building the ML models

```
control <- trainControl(method = "cv", number = 10)

model1 <- train(stars~.,data = train,method = "knn",metric = "Accuracy",tuneLength = 5,trCont
rol = control, preProcess="scale")
model1
```

```
## k-Nearest Neighbors
##
## 6166 samples
##    6 predictor
##    5 classes: '1', '2', '3', '4', '5'
##
## Pre-processing: scaled (6)
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 5550, 5550, 5548, 5550, 5550, 5547, ...
## Resampling results across tuning parameters:
##
##   k   Accuracy   Kappa
##    5  0.3554976  0.08969417
##    7  0.3639282  0.09612011
##    9  0.3627918  0.09132157
##   11  0.3687854  0.09668969
##   13  0.3666827  0.09179807
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was k = 11.
```

```
model2 <- train(stars~.,data = train,method = "rpart",metric = "Accuracy",tuneLength = 5,trCo
ntrol = control, preProcess="scale")
model2
```

```
## CART
##
## 6166 samples
##    6 predictor
##    5 classes: '1', '2', '3', '4', '5'
##
## Pre-processing: scaled (6)
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 5552, 5549, 5547, 5550, 5548, 5549, ...
## Resampling results across tuning parameters:
##
##   cp           Accuracy   Kappa
##   0.002013592  0.4012399  0.13887546
##   0.002391140  0.3997764  0.13750934
##   0.011074755  0.4031850  0.14368579
##   0.014346841  0.3984846  0.14578713
##   0.031084823  0.3592281  0.02700401
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was cp = 0.01107475.
```

```
control <- trainControl(method = "repeatedcv", number = 5)
model3 <- train(stars~.,data = train,method = "rf",metric = "Accuracy",tuneLength = 5,trContr
ol = control, preProcess="scale")
model3
```

```
## Random Forest
##
## 6166 samples
##    6 predictor
##    5 classes: '1', '2', '3', '4', '5'
##
## Pre-processing: scaled (6)
## Resampling: Cross-Validated (5 fold, repeated 1 times)
## Summary of sample sizes: 4932, 4933, 4932, 4934, 4933
## Resampling results across tuning parameters:
##
##   mtry  Accuracy   Kappa
##   2     0.3806334  0.1109955
##   3     0.3710668  0.1076365
##   4     0.3647428  0.1027906
##   5     0.3623073  0.1026418
##   6     0.3621466  0.1049788
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was mtry = 2.
```

A confusion matrix is built for all the above models

Confusion Matrix

```
aa <- predict(model1,test)
confusionMatrix(aa,test$stars)
```

```
## Confusion Matrix and Statistics
## 
##           Reference
## Prediction   1   2   3   4   5
##          1  75  45  24  24  24
##          2  38  30  16  19  19
##          3  23  31  38  48  47
##          4  44  85 167 471 394
##          5  22  46 144 376 391
## 
## Overall Statistics
## 
##                Accuracy : 0.3805          
##                  95% CI : (0.362, 0.3994) 
##     No Information Rate : 0.3552          
##     P-Value [Acc > NIR] : 0.003549        
## 
##                   Kappa : 0.1162          
##  Mcnemar's Test P-Value : < 2.2e-16       
## 
## Statistics by Class:
## 
##                      Class: 1 Class: 2 Class: 3 Class: 4 Class: 5
## Sensitivity           0.37129  0.12658  0.09769   0.5021   0.4469
## Specificity           0.95203  0.96173  0.93384   0.5948   0.6670
## Pos Pred Value        0.39062  0.24590  0.20321   0.4057   0.3994
## Neg Pred Value        0.94814  0.91782  0.85697   0.6845   0.7088
## Prevalence            0.07649  0.08974  0.14729   0.3552   0.3313
## Detection Rate        0.02840  0.01136  0.01439   0.1783   0.1480
## Detection Prevalence  0.07270  0.04619  0.07081   0.4396   0.3707
## Balanced Accuracy     0.66166  0.54416  0.51576   0.5485   0.5570
```

```
#### Model 1 is a KNN model and has an average accuraccy of 0.3851. Overall the sensitivity s
cores are really low. The accuracy score for each individual class is fair

aa <- predict(model2,test)
confusionMatrix(aa,test$stars)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   1   2   3   4   5
##          1 101  63  34  24  25
##          2   0   0   0   0   0
##          3   0   0   0   0   0
##          4  98 158 278 546 414
##          5   3  16  77 368 436
##
## Overall Statistics
##
##                Accuracy : 0.4101
##                  95% CI : (0.3912, 0.4291)
##     No Information Rate : 0.3552
##     P-Value [Acc > NIR] : 2.989e-09
##
##                   Kappa : 0.1312
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: 1 Class: 2 Class: 3 Class: 4 Class: 5
## Sensitivity           0.50000  0.00000   0.0000   0.5821   0.4983
## Specificity           0.94014  1.00000   1.0000   0.4433   0.7373
## Pos Pred Value        0.40891      NaN      NaN   0.3655   0.4844
## Neg Pred Value        0.95781  0.91026   0.8527   0.6582   0.7478
## Prevalence            0.07649  0.08974   0.1473   0.3552   0.3313
## Detection Rate        0.03824  0.00000   0.0000   0.2067   0.1651
## Detection Prevalence  0.09353  0.00000   0.0000   0.5657   0.3408
## Balanced Accuracy     0.72007  0.50000   0.5000   0.5127   0.6178
```

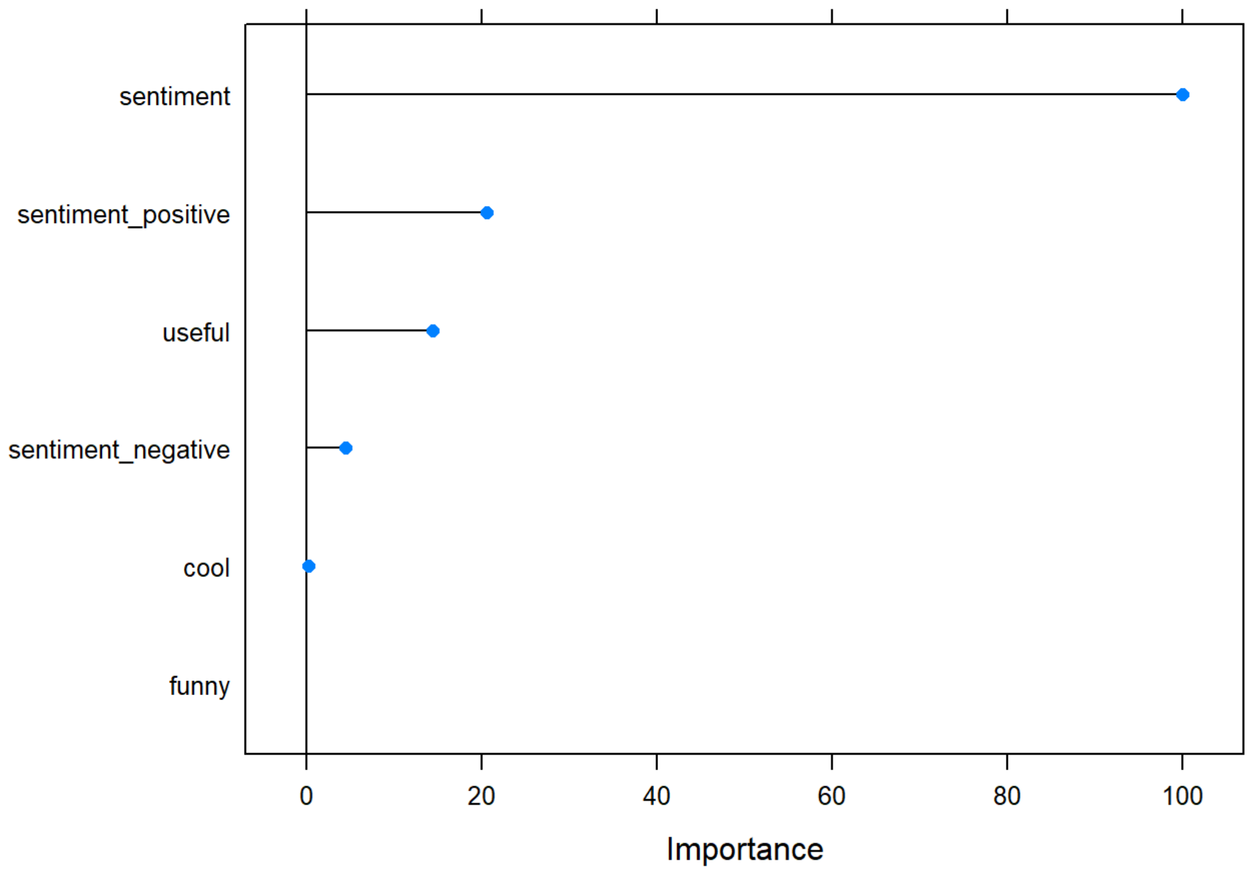
```
#### Model 2 is a Desicion Tree model and has an average accuracy of 0.4089. But in the model
 there is no prediction for 3 star ratings. Sensitivity for 3 stars is zero. For this reason
 this model cannot be relied upon even if it has better accuracy.

aa <- predict(model3,test)
confusionMatrix(aa,test$stars)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   1   2   3   4   5
##          1  92  47  35  27  20
##          2  27  35  21  22  22
##          3  19  15  17  36  28
##          4  46 108 217 512 412
##          5  18  32  99 341 393
##
## Overall Statistics
##
##                Accuracy : 0.3972
##                  95% CI : (0.3785, 0.4162)
##     No Information Rate : 0.3552
##     P-Value [Acc > NIR] : 4.122e-06
##
##                   Kappa : 0.1363
##  Mcnemar's Test P-Value : < 2.2e-16
##
## Statistics by Class:
##
##                      Class: 1 Class: 2 Class: 3 Class: 4 Class: 5
## Sensitivity           0.45545  0.14768 0.043702   0.5458   0.4491
## Specificity           0.94711  0.96173 0.956483   0.5402   0.7225
## Pos Pred Value        0.41629  0.27559 0.147826   0.3954   0.4451
## Neg Pred Value        0.95455  0.91965 0.852732   0.6835   0.7258
## Prevalence            0.07649  0.08974 0.147293   0.3552   0.3313
## Detection Rate        0.03484  0.01325 0.006437   0.1939   0.1488
## Detection Prevalence  0.08368  0.04809 0.043544   0.4903   0.3343
## Balanced Accuracy     0.70128  0.55470 0.500092   0.5430   0.5858
```

Model 3 is a Random Forest model and has an average accuracy of 0.3959. This model also has better sensitivity and specificity scores than the others.

The important variables are found using the VarImp plot

```
plot(caret::varImp(model3))
```



This model says that the sentiment scores are highly important than the rest of the variables. The intense amount of missing values in cool, funny and useful have contributed nothing to the model. Most of the predictions were made with the help of the sentiment scores.If there are more information about the reviwe, the cuisine, hotel location, etc, they could have acted as better predictors of the ratings than just the reviwes.