Chapter4_example1_HSY.R

User

Sat Sep 29 02:51:15 2018

```
setwd("E:WWBITAminWWMachine Learning with R, Second Edition_CodeWWChapter 04")

## importing datasets
mandrill = read.csv("Mandrill.csv", header=T)
other = read.csv("Other.csv", header=T)

## Q1.
mandrill['class'] = rep('app', 150)
other['class'] = rep('other', 150)
total = rbind(mandrill, other)

## Q2.
str(total$class)
```

```
##  chr [1:300] "app" "app" "app" "app" "app" "app" "app" "app" "app" ...
```

```
total$class = as.factor(total$class)
str(total$class)
```

```
##  Factor w/ 2 levels "app","other": 1 1 1 1 1 1 1 1 1 1 ...
```

```
## Q3.
library(tm)
```

```
## Loading required package: NLP
```

```
total_corpus = VCorpus(VectorSource(total$Tweet))
as.character(total_corpus[[1]])  #??
```

```
## [1] "[blog] Using Nullmailer and Mandrill for your Ubuntu Linux server outboud mail:  htt
p://bit.ly/ZjH0k7  #plone"
```

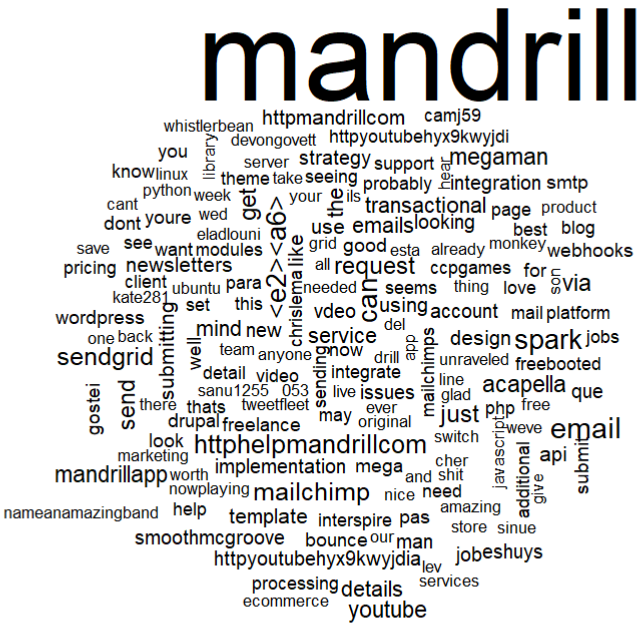
```
## Q4.
x = tm_map(total_corpus, content_transformer(tolower))
x = tm_map(x, removePunctuation)
x = tm_map(x, removeNumbers)
x = tm_map(x, stripWhitespace)
x = tm_map(x, removeWords, c(stopwords('english'), 'will', 'just', 'get', 'mandrill') )
x = tm_map(x, stemDocument, language = 'english')

## Q5.
total_dtm = DocumentTermMatrix(x)

## Q6.
set.seed(1004)
N = nrow(total)
sampling = sample(N, N*0.7)
tweet_train = total_dtm[sampling, ]
tweet_test = total_dtm[-sampling, ]
tweet_train_labels = total[sampling, ]$class
tweet_test_labels = total[-sampling, ]$class

## Q7.
library(wordcloud)
```

```
## Loading required package: RColorBrewer
```

```
app = subset(total, class='app')
other = subset(total, class='other')
wordcloud(app$Tweet)
```

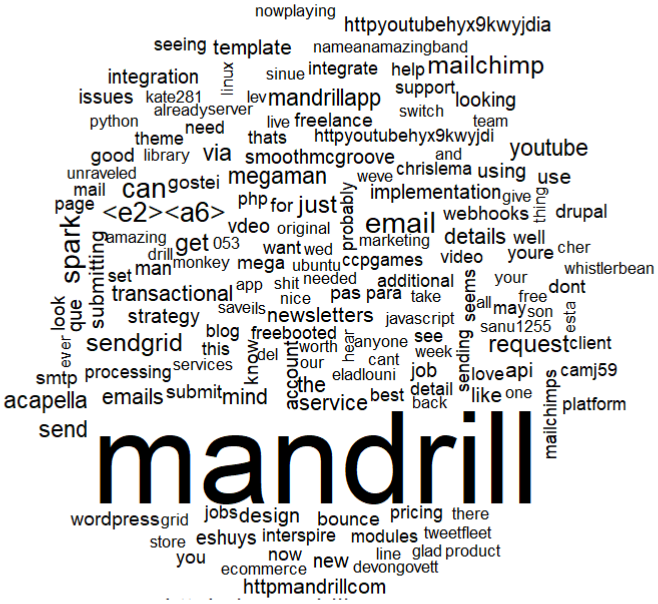
```
## Warning in tm_map.SimpleCorpus(corpus, tm::removePunctuation):
## transformation drops documents
```

```
## Warning in tm_map.SimpleCorpus(corpus, function(x) tm::removeWords(x,
## tm::stopwords())): transformation drops documents
```




```
wordcloud(other$Tweet)
```

```
## Warning in tm_map.SimpleCorpus(corpus, tm::removePunctuation):
## transformation drops documents

## Warning in tm_map.SimpleCorpus(corpus, tm::removePunctuation):
## transformation drops documents
```




```
## Q8.
tweet_freq_words = findFreqTerms(tweet_train, 2)
str(tweet_freq_words)
```

```
##  chr [1:233] "acapella" "account" "addit" "adicionei" "alreadi" ...
```

```
tweet_freq_train = tweet_train[ , tweet_freq_words]
tweet_freq_test = tweet_test[ , tweet_freq_words]
convert_counts = function(x) {
  x = ifelse(x>0, "Yes", "No")
}
tweet_train = apply(tweet_freq_train, MARGIN = 2,  convert_counts)
tweet_test = apply(tweet_freq_test, MARGIN = 2, convert_counts)

library(e1071)
tweet_classifier = naiveBayes(tweet_train, tweet_train_labels)

## Q9.
tweet_test_pred = predict(tweet_classifier, tweet_test)

## evaluating model performance.
library(gmodels)
CrossTable(tweet_test_pred, tweet_test_labels, prop.chisq=F, prop.t=F,
           dnn=c("predicted", "actual"))
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  90
##
##
##              | actual
##    predicted |       app |     other | Row Total |
## -------------|-----------|-----------|-----------|
##          app |        35 |         4 |        39 |
##              |     0.897 |     0.103 |     0.433 |
##              |     0.875 |     0.080 |           |
## -------------|-----------|-----------|-----------|
##        other |         5 |        46 |        51 |
##              |     0.098 |     0.902 |     0.567 |
##              |     0.125 |     0.920 |           |
## -------------|-----------|-----------|-----------|
## Column Total |        40 |        50 |        90 |
##              |     0.444 |     0.556 |           |
## -------------|-----------|-----------|-----------|
##
##
```