

Watson Lab Sentiment Analysis

Surabhi Chouhan

January 10, 2017

Get access to IBM Watson Lab

```
#to use install_github() to install packages hosted on GitHub
library("devtools")

#package used to use sentiment analysis and other features of Alchemy
Language of Watson Lab
devtools::install_github("ColumbusCollaboratory/cognizer")

## Skipping install of 'cognizer' from a github remote, the SHA1 (4d368b5c)
has not changed since last install.

library("cognizer")

#code used to set R environment and use the AlchemyAPI key
r_env <- file.path(normalizePath("~"), ".Renviron")
if (!file.exists(r_env)) file.create(r_env)

#Getting authentication for using AlchemyAPI
SERVICE_API_KEY = "e806f3bc37a652d9dd7c29af9355a64a4d547917"
```

Sentiment analysis of SBA text using Alchemy language of Watson Lab

```
#Step 2: Get the sentiment scores for SBA speech

SBA <- readLines("D:/Surabhi docs/portfolio/SBA-speech.txt")

#getting the sentiment score using the inbuilt function from Watson Lab
result <- text_sentiment(SBA,"e806f3bc37a652d9dd7c29af9355a64a4d547917")

#Divide the text into 4 parts by sentence

splitSBA = split(SBA,sample(rep(1:2:3:4,600))) #where 538 is the number of
words in the speech

set.seed(10)
result1 <-
text_sentiment(splitSBA$`1`, "e806f3bc37a652d9dd7c29af9355a64a4d547917")
```

```

result2 <-
text_sentiment(splitSBA$`2`, "e806f3bc37a652d9dd7c29af9355a64a4d547917")

result3 <-
text_sentiment(splitSBA$`3`, "e806f3bc37a652d9dd7c29af9355a64a4d547917")

result4 <-
text_sentiment(splitSBA$`4`, "e806f3bc37a652d9dd7c29af9355a64a4d547917")

#The following statements gets just the score from the data frame and stores it in a vector
result1[[1]]$docSentiment$score
## NULL

result2[[2]]$docSentiment$score
## [1] "-0.625445"

result3[[3]]$docSentiment$score
## NULL

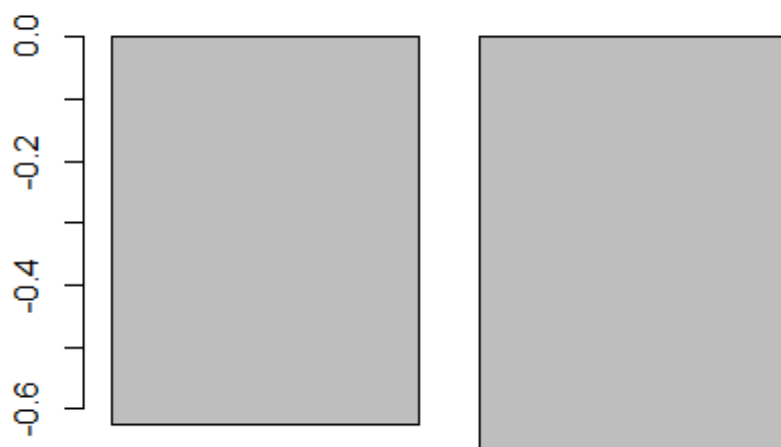
result4[[2]]$docSentiment$score
## [1] "-0.669016"

allResults <-
as.numeric(c(result1[[1]]$docSentiment$score, result2[[2]]$docSentiment$score,
result3[[3]]$docSentiment$score, result4[[2]]$docSentiment$score))

#plotting the sentiment score for each quarter of speech on a barplot
barplot(allResults, main="Sentiment analysis")

```

Sentiment analysis



#Emotion analysis using Alchemy language of Watson Lab

#Step 4: Try to explore more functions (i.e., emotion analysis) of Watson through R

#getting the emotion analysis using the inbuilt function from Watson Lab

```
SBA_emotion <- text_emotion(SBA,"e806f3bc37a652d9dd7c29af9355a64a4d547917")  
SBA_emotion[[15]]$docEmotions
```

```
## $anger  
## [1] "0.40958"  
##  
## $disgust  
## [1] "0.208747"  
##  
## $fear  
## [1] "0.093455"  
##  
## $joy  
## [1] "0.00113"  
##  
## $sadness  
## [1] "0.496388"
```

#split the speech into 4 parts

```
splitSBA1 = split(SBA,sample(rep(1:2:3:4,600)))
```

```

set.seed(10)
result_emo1 <-
text_emotion(splitSBA$`1`, "e806f3bc37a652d9dd7c29af9355a64a4d547917")

result_emo2 <-
text_emotion(splitSBA$`2`, "e806f3bc37a652d9dd7c29af9355a64a4d547917")

result_emo3 <-
text_emotion(splitSBA$`3`, "e806f3bc37a652d9dd7c29af9355a64a4d547917")

result_emo4 <-
text_emotion(splitSBA$`4`, "e806f3bc37a652d9dd7c29af9355a64a4d547917")

result_emo1[[4]]$docEmotions

## NULL

result_emo2[[2]]$docEmotions

## $anger
## [1] "0.40958"
##
## $disgust
## [1] "0.208747"
##
## $fear
## [1] "0.093455"
##
## $joy
## [1] "0.00113"
##
## $sadness
## [1] "0.496388"

result_emo3[[1]]$docEmotions

## $anger
## [1] "0.180649"
##
## $disgust
## [1] "0.379431"
##
## $fear
## [1] "0.101488"
##
## $joy
## [1] "0.006845"
##
## $sadness
## [1] "0.457807"

```