

Question

During this semester you will have two projects. one of them before the mid-term another one before final. You can be grouped in a maximum of 2 persons. You must write your project in R. I emphasize that any other languages are not acceptable. I will not give any score for incomplete projects. Also, Really if I understand that you cheated and copy again this time you deserve a big zero. Project 1: retrieve posts of a camera manufacturer’s page and its comments on facebook to see how well readers received it. our aim is to analyses comments to find the rate of positive comments. You also should fetch data from the facebook. Good luck!

Answer

#libraries

library(tuber)

library(tm)

library(SentimentAnalysis)

library(textreg)

library(ggplot2)

library(wordcloud)

#global variables

app\_id <- ""

app\_secret <- ""

video\_id <- ""

file = "comments.csv"

#functions

connectToApi <- **function**(){

  yt\_oauth(app\_id, app\_secret)

}

fetchData <- **function**(){

**return**(get\_all\_comments(video\_id = video\_id))

}

writeCsv <- **function**(data, file){

  write.csv(data, file=file)

}

readCsv <- **function**(file){

**return**(read.csv(file = file, stringsAsFactors = FALSE))

}

clearData <- **function**(data){

  sms\_corpus <- VCorpus(VectorSource(data**$**textOriginal))

  sms\_corpus\_clean <- tm\_map(sms\_corpus, content\_transformer(tolower))

  sms\_corpus\_clean <- tm\_map(sms\_corpus\_clean, removeNumbers)

  sms\_corpus\_clean <- tm\_map(sms\_corpus\_clean, removePunctuation)

  sms\_corpus\_clean <- tm\_map(sms\_corpus\_clean, removeWords, c("the", "and",stopwords("english")))

  sms\_corpus\_clean <- tm\_map(sms\_corpus\_clean, stripWhitespace)

**return**(sms\_corpus\_clean)

}

sentimentData <- **function**(text){

  sentiment <- analyzeSentiment(text)

  print(sentiment)

**return**(cbind(sentiment**$**WordCount, convertToBinaryResponse(sentiment)**$**SentimentGI))

}

calculatePositivityFrequency <- **function**(words){

  pos.counter <- 0

**for**(i **in** 1**:**nrow(words)){

**if**(words[i,2] == 2 **&&** **!**is.na(words[i,2])){

      pos.counter <- pos.counter + words[i,1]

    }

  }

**return**(pos.counter)

}

calculateNegativityFrequency <- **function**(words){

  neg.counter <- 0

**for**(i **in** 1**:**nrow(words)){

**if**(words[i,2] == 1 **&&** **!**is.na(words[i,2])){

      neg.counter <- neg.counter + words[i,1]

    }

  }

**return**(neg.counter)

}

getRate <- **function**(frequency, count){

**if**(count == 0){

**return**(-1)

  }

**return**(frequency / count \* 100)

}

getlength <- **function**(words){

  lengthOfWords = 0

**for**(count **in** 1**:**nrow(words)){

    lengthOfWords <- lengthOfWords + words[count,1]

  }

**return**(lengthOfWords)

}

main <- **function**(){

  #connectToApi()

  #data <- fetchData()

  #writeCsv(data, file)

  data <- readCsv(file)

  data <- convert.tm.to.character(clearData(data))

  data <- na.omit(data)

  data.array <- as.matrix(data)

  words <- sentimentData(data)

  pos.frequency <- calculatePositivityFrequency(words)

  neg.frequency <- calculateNegativityFrequency(words)

  lengthOfWords <- getlength(words)

  pos.rate <- getRate(pos.frequency, lengthOfWords)

  neg.rate <- getRate(neg.frequency, lengthOfWords)

  print(paste("positivity rate: " , pos.rate))

  print(paste("negativity rate: " , neg.rate))

  wordcloud(

    words=data,

    colors = brewer.pal(5, 'Dark2'),

    scale=c(3, 0.4),

    max.words = 50,

    rot.per = 0.2,

    min.freq = 5

  )

  hist(

     words[,2],

     col= rainbow(5),

     xlim=c(1,2),

     breaks = 2,

     main="Histogram of pos and neg rate",

     xlab="pos=1, neg=2",

     las = 1

   )

}

main()