Name: sagar sonawane PRN No : 0220190030

Branch: Computer Roll No.: 49

***Lab Assignment 18/10/2020***

'

install.packages("twitteR", dependencies = TRUE)

install.packages("RCurl")

install.packages("bitops")

install.packages("base64enc")

install.packages("httpuv")

install.packages("tm")

install.packages("wordcloud")

install.packages("stringr")

'

libs <- c("twitteR","RCurl","bitops","base64enc","httpuv","tm","wordcloud","stringr")

# array of all packages

lapply(libs, require, character.only = TRUE) #install all package at once

install.packages("qdap")

library(dplyr)

library(qdap)

text<- "Text writing usually involves the process of structuring

the input text.The overarching goals is,essentially

to turn text into data for analysis,via application of natural

language processing(NPL) and analytical methods"

str(text)

frequent\_term = freq\_terms(text,4)

frequent\_term

plot(frequent\_term)

example\_text = data.frame(doc\_id = c(1,2,3),

text = c("Text mining is a great time.",

"Text analysis provides insights",

"qdap and tm are used in text mining"),

author = c("Author1", "Author2", "Author3"),

date = c("1514953399", "1514866998", "1514780598")

)

example\_text

doc\_id text author date

1 1 Text mining is a great time. Author1 1514953399

2 2 Text analysis provides insights Author2 1514866998

3 3 qdap and tm are used in text mining Author3 1514780598

'

vector\_text = c("Text mining is a great time.",

"Text analysis provides insights",

"qdap and tm are used in text mining")

# Compare corpus from dataframe and vector

# Create a DataframeSource: df\_source

df\_source = DataframeSource(example\_text)

# Create a VectorSource: vec\_source

vec\_source = VectorSource(vector\_text)

# Convert df\_source to a corpus: df\_corpus

df\_corpus = VCorpus(df\_source)

# Convert vec\_source to a corpus: vec\_corpus

vec\_corpus = VCorpus(vec\_source)

# Examinedf\_corpus

df\_corpus

'

<<VCorpus>>

Metadata: corpus specific: 0, document level (indexed): 2

Content: documents: 3

'

# Compare the number of documents in the vector source

# Examinevec\_corpus

vec\_corpus

'

<<VCorpus>>

Metadata: corpus specific: 0, document level (indexed): 0

Content: documents: 3

'

# Examinedf\_corpus metadata

meta(df\_corpus)

'

author date

1 Author1 1514953399

2 Author2 1514866998

3 Author3 1514780598

'

# Compare metadata in the vector corpus

# Examinevec\_corpus metadata

meta(vec\_corpus)

#data frame with 0 columns and 3 rows

coffee\_tweets\_ord = read.csv("D:/R Programming/Dataset/coffee.csv", stringsAsFactors = FALSE)

coffee\_tweets = coffee\_tweets\_ord

dim(coffee\_tweets)

glimpse(coffee\_tweets)

View(head(coffee\_tweets))

View(coffee\_tweets)

str(coffee\_tweets)

summary(coffee\_tweets)

# We are interested in the text column for text mining

coffee\_tweets = coffee\_tweets$text

str(coffee\_tweets)

head(coffee\_tweets, 4)

"

STEP 1:

Create a source object from the coffee\_tweets vector

Call this coffee\_sourse

"

coffee\_source = VectorSource(coffee\_tweets)

class(coffee\_source)

# Make a volatile corpus: coffee\_corpus

coffee\_corpus = VCorpus(coffee\_source)

coffee\_corpus

# Printing of 15th corpus

coffee\_corpus[[15]]

# Print the content of 15th tweet in coffee\_corpus

content(coffee\_corpus[[15]])

# view 10th one

content(coffee\_corpus[[10]])

example\_text = data.frame(doc\_id = c(1, 2, 3))

# Create the object: text

text<- "<b>She</b> woke up at 6 A.M. It\'s so early! She was only 10% awake and began drinking coffee in front of her computer."

# Make lowercase

tolower(text)

"

[1] '<b>she</b> woke up at 6 a.m. it's so early!

she was only 10% awake and began drinking coffee in front

of her computer.'

"