week1 Capstone

Nirmal Ghimire

1/4/2021

Tasks to accomplish

Tokenization - identifying appropriate tokens such as words, punctuation, and numbers. Writing a function that takes a file as input and returns a tokenized version of it.

Profanity filtering - removing profanity and other words you do not want to predict.

Tips, Tricks, and Hints

Loading the data in.

a. This dataset is fairly large. We emphasize that you don't necessarily need to load the entire dataset in to build your algorithms (see point 2 below).
b. At least initially, you might want to use a smaller subset of the data. Reading in chunks or lines using R's **readLines** or **scan** functions can be useful.
c. You can also loop over each line of text by embedding readLines within a for/while loop, but this may be slower than reading in large chunks at a time.
d. Reading pieces of the file at a time will require the use of a file connection in R. For example, the following code could be used to read the first few lines of the English Twitter

- dataset:con <- file("en_US.twitter.txt", "r")

readLines(con, 1) ## Read the first line of text

readLines(con, 1) ## Read the next line of text

readLines(con, 5) ## Read in the next 5 lines of text

close(con) ## It's important to close the connection when you are done.

See the connections help page for more information.

Sampling.

To reiterate, to build models you don't need to load in and use all of the data. Often relatively few randomly selected rows or chunks need to be included to get an accurate approximation to results that would be obtained using all the data. Remember your inference class and how a representative sample can be used to infer facts about a population.

a. You might want to create a separate sub-sample dataset by reading in a random subset of the original data and writing it out to a separate file.

b. That way, you can store the sample and not have to recreate it every time.
c. You can use the rbinom function to “flip a biased coin” to determine whether you sample a line of text or not.

Quiz 1: Getting Started

1. The en_US.blogs.txt file is how many megabytes?

```
givendata<-(file.info("C:/Users/nirma/Documents/Coursera/Data Science/Data
Science Capstone/final/en_US/en_US.blogs.txt")$size)/1024/1024
sprintf("The en_US.blogs.txt file is: %s Megabytes", givendata)

## [1] "The en_US.blogs.txt file is: 200.424207687378 Megabytes"
```

2. The en_US.twitter.txt|en_US.twitter.txt has how many lines of text?

```
twitterdata<-file("C:/Users/nirma/Documents/Coursera/Data Science/Data
Science Capstone/final/en_US/en_US.twitter.txt", open="rb")
total_lines<- 0L
while(length(chunk<-readBin(twitterdata,"raw", 65536))>0){
  total_lines<-total_lines+sum(chunk==as.raw(10L))
}
close(twitterdata)
sprintf("The en_US.twitter.txt file has: %s Lines", total_lines)

## [1] "The en_US.twitter.txt file has: 2360148 Lines"
```

3. What is the length of the longest line seen in any of the three en_US data sets?

```
blogsdata<-file("C:/Users/nirma/Documents/Coursera/Data Science/Data Science
Capstone/final/en_US/en_US.blogs.txt", open="rb")
twitterdata<-file("C:/Users/nirma/Documents/Coursera/Data Science/Data
Science Capstone/final/en_US/en_US.twitter.txt", open="rb")
newsdata<-file("C:/Users/nirma/Documents/Coursera/Data Science/Data Science
Capstone/final/en_US/en_US.news.txt", open="rb")
#Reading Lines
blogsdata_lines<-readLines(blogsdata)
close(blogsdata)
blogsdata_L<-summary(nchar(blogsdata_lines))[6]
blogsdata_L

##  Max.
## 40835

twitterdata_lines<-readLines(twitterdata)

## Warning in readLines(twitterdata): line 167155 appears to contain an
embedded
## nul
## Warning in readLines(twitterdata): line 268547 appears to contain an
embedded
## nul

## Warning in readLines(twitterdata): line 1274086 appears to contain an
embedded
## nul

## Warning in readLines(twitterdata): line 1759032 appears to contain an
embedded
## nul

close(twitterdata)
twitterdata_L<-summary(nchar(twitterdata_lines))[6]
twitterdata_L

## Max.
##  213

newsdata_lines<-readLines(newsdata)
close(newsdata)
newsdata_L<-summary(nchar(newsdata_lines))[6]
newsdata_L

##  Max.
## 11384
```

4.In the en_US twitter data set, if you divide the number of lines where the word "love" (all lowercase) occurs by the number of lines the word "hate" (all lowercase) occurs, about what do you get?

```
lovehate<-length(grep("love", twitterdata_lines))/length(grep("hate",
twitterdata_lines))
sprintf("We get around: %s", lovehate)

## [1] "We get around: 4.10859156202006"
```

5. The one tweet in the en_US twitter data set that matches the word "biostats" says what?

```
biostat_tweet<-grep("biostats", twitterdata_lines, value=TRUE)
biostat_tweet

## [1] "i know how you feel.. i have biostats on tuesday and i have yet to
study =/"
```

6. How many tweets have the exact characters "A computer once beat me at chess, but it was no match for me at kickboxing". (I.e. the line matches those characters exactly.)

```
tweet_match<-grep("A computer once beat me at chess, but it was no match for
me at kickboxing", twitterdata_lines)
tweet_match
```

```
## [1]  519059  835824 2283423
```