IST 387 HW 11

```
# Enter your name here: Ezra Cohen
```

Attribution statement: (choose only one and delete the rest)

```
# 1. I did this homework by myself, with help from the book and the professor.
```

Text mining plays an important role in many industries because of the prevalence of text in the interactions between customers and company representatives. Even when the customer interaction is by speech, rather than by chat or email, speech to text algorithms have gotten so good that transcriptions of these spoken word interactions are often available. To an increasing extent, a data scientist needs to be able to wield tools that turn a body of text into actionable insights. In this homework, we explore a real **City of Syracuse dataset** using the **quanteda** package. Make sure to install the quanteda package before following the steps below:

Part 1: Load and visualize the data file

A. Take a look at this article: https://samedelstein.medium.com/snowplow-naming-contest-data-2dcd38272caf and write a comment in your R script, briefly describing what it is about.

```
#it is about a contest for naming snowplows
```

B. Copy the data from the following URL to a dataframe called **df**: https://ist387.s3.us-east-2.amazonaws.com/data/snowplownames.csv

```
df<-data.frame(read_csv("https://ist387.s3.us-east-2.amazonaws.com/data/snowplownames.csv"))
```

C. Inspect the **df** dataframe – which column contains an explanation of the meaning of each submitted snowplow name? Transform that column into a **document-feature matrix**, using the **corpus()** and **dfm()** functions. Do not forget to **remove stop words**.

```
View(df)
dfCorpus <- corpus(df$meaning)
dfDFM <- dfm(dfCorpus,remove_punct=TRUE,remove=stopwords("english"))
```

D. Plot a **word cloud** where a word is only represented if it appears **at least 2 times** in the corpus. **Hint:** use **textplot_wordcloud()**:

```
textplot_wordcloud(dfDFM,min_count=2)
```

E. Next, **increase the minimum count to 10**. What happens to the word cloud? **Explain in a comment**.

```
textplot_wordcloud(dfDFM,min_count=10)
#the wordcloud is much smaller because words need to appear more often to make it into this one
```

F. What are the top words in the word cloud? Explain in a brief comment.

```
#snow plow name syracuse plows and salt which all have to do with the topic as a whole
```

Part 2: Create a sorted list of word counts from the reviews

G. Create a **named list of word counts by frequency**.
Hint: use the functions as.matrix() and colSums()

```
sorteddfm <- as.matrix(dfDFM)
wordcounts <- colSums(sorteddfm)
wordcounts
```

H. Explain in a comment what you observed in the sorted list of word counts.

```
#it says how many time each word occured in total
```

Part 3: Match the review words with positive and negative words

I. Read in the list of positive words (using the scan() function), as well as the negative words list:
https://ist387.s3.us-east-2.amazonaws.com/data/positive-words.txt

https://ist387.s3.us-east-2.amazonaws.com/data/negative-words.txt

There should be 2006 positive words and 4783 negative words, so you may need to clean up these lists a bit.

```
poswords <- scan("https://ist387.s3.us-east-2.amazonaws.com/data/positive-words.txt",character(0),sep=";")
poswords <- poswords[-1:-216]
negwords <- scan("https://ist387.s3.us-east-2.amazonaws.com/data/negative-words.txt",character(0),sep=";")
negwords <- negwords[-1:-220]
```

J. Here's a code example for matching the words from the name explanations (stored in **wordCounts**) to the list of positive words (stored in **posWords**):

```
matchedP <- match(names(wordcounts), poswords, nomatch = 0)
```

```
Error in match(names(wordCounts), posWords, nomatch = 0): object 'wordCounts' not found
Traceback:


1. match(names(wordCounts), posWords, nomatch = 0)
```

Create a similar line of code to match the name explanations to the negative words.

```
matchedN <- match(names(wordcounts), negwords, nomatch = 0)
```

K. Examine the contents of **matchedP**. What does each non-zero entry contain? How does that relate to **wordCounts** and **posWords**?

```
matchedP[which(matchedP!=0)]
#it contains the index of the matching word in poswords
```

L. Use R to print out the positive words in the name explanation variable in **df**.

```
poswords[matchedP[which(matchedP!=0)]]
```

M. Use R to print out the total number of positive words in the name explanation variable in **df**.

```
length(matchedP[which(matchedP!=0)])
```

N. Repeat that process for the negative words you matched. Which negative words were in the name explanation variable, and what is their total number?

```
matchedN[which(matchedN!=0)]
negwords[matchedN[which(matchedN!=0)]]
length(matchedN[which(matchedN!=0)])
```

O. Write a comment describing what you found after matching positive and negative words. Which group is more common in this dataset?

```
#There are way more positive words than negative words
```