JHU Data Science Capstone Project - Milestone Report

Sunday, March 29, 2015

# Synopsis

This report explores the corpus, HC Corpora, from <http://www.corpora.heliohost.org/> in preparation for the John Hopkins Data Science Capstone project. The project is a Shiny app that takes as input a phrase (multiple words) in a text box input and outputs a prediction of the next word. This document explains only the major features identified of the data including summary statistics about the data sets. It also reports any interesting findings that we amassed so far.

In addition, the report elaborates our goals for the eventual app and algorithm and briefly summarize our plans for creating the prediction algorithm and Shiny app. It also serves as a basis for collecting feedback on our plans for creating a prediction algorithm and Shiny app.

# Data Processing

## Loading packages

We used different R packages for producing report. More specifically:  
\* For text manipulation, we used: RWeka, stringi  
\* For text mining, we used: SnowballC, tm  
\* For data visualization, we used: ggplot2

## Downloading data

[The HC Corpora](https://d396qusza40orc.cloudfront.net/dsscapstone/dataset/Coursera-SwiftKey.zip) where downloaded from www.corpora.heliohost.org . See the readme file at <http://www.corpora.heliohost.org/aboutcorpus.html> for details on the corpora available. The files have been language filtered but may still contain some foreign text.

The file downloaded had a large size of 548MB. After unzipping the file, we find the following directories:

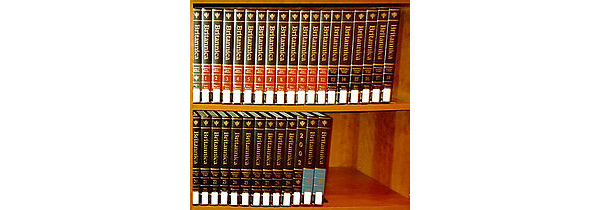
## [1] "de\_DE" "en\_US" "fi\_FI" "ru\_RU"

Each of them represent Corpora for a specific language. We will use English Corpora here for our proficiency with the English language. If we list the files in the english corpora directory, we see that it contains files for news, blogs and twitter.

## [1] "en\_US.blogs.txt" "en\_US.news.txt" "en\_US.twitter.txt"

## Loading data

When loading all of the english Corpora in the data file, we are actually loading around 151 million words in 2 million lines. That's about four times the size of the Encyclopedia Britannica.



## Data Features and Summary Statistics

To get a better sense of the size of this data, the following summary statistics elaborates the main features of the corpora.

## filename size\_MB total\_words lines mean\_words  
## 1 en\_US.blogs.txt 200.4242 38154238 899288 42.42716  
## 2 en\_US.news.txt 196.2775 35010782 1010242 34.65584  
## 3 en\_US.twitter.txt 159.3641 2136398 167155 12.78094  
## 4 Total 556.0658 75301418 2076685 34.65582

## Data Sampling

Beacause the data size is too large which has resulted in many slowdowns and freezes in the computer while exploring the data, I have taken a sample that constitutes 10% of the original data. 10% of each of the files downloaded for the english corpora, the number of lines of the sample is around 200,000 lines verses 2,000,000 lines for the original corpora.

## Data Cleaning

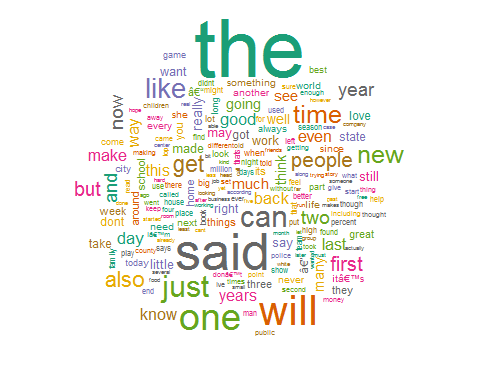
Before delving into tokenization of the corpora, i.e. extracting the unique n-gram words or phrases, we will clean the corpora through the following transofrmations:  
 1- Converting the corpora text into UTF-8 encoding.  
 2- Removing numbers.  
 3- Removing unnecessary punctuation.  
 4- Optionally removing English stopwords, we might cancel this step later as we progress as it may affect the perfromance of our predictive model.  
 5- Optionally removing profanity words, we might cancel this step later as we progress as it may affect the perfromance of our predictive model. 6- Converting all alphabetical characters into lowercase.  
 7- Removing extra white space such as tab, etc...  
 8- Stemming the different words, i.e. removing any suffixes such ing, er, s, etc...  
 9- Converting the text into plain text document.

## Corpora Tokenization

### Most Frequent Unigrams (1 word)

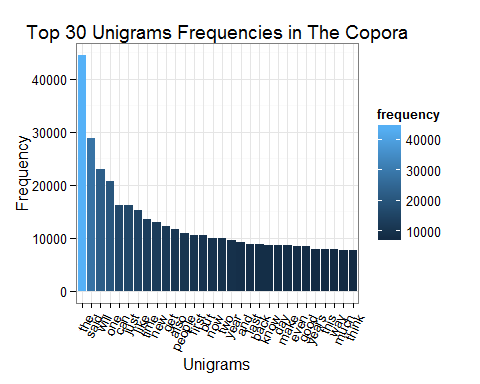
#### Bigrams Word Cloud

The word cloud shows the most common unigrams larger in scale.



#### Unigrams Histogram

The histogram shows the different frequencies of the top unigrams.



### Most Frequent Bigrams (2 words)

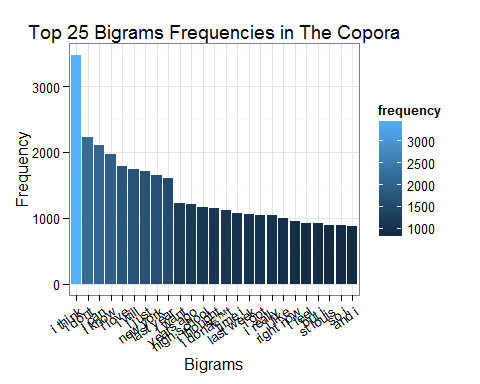
#### Bigrams Word Cloud

The word cloud shows the most common bigrams larger in scale.



#### Bigrams Histogram

The histogram shows the different frequencies of the top bigrams.



### Most Frequent Trigrams (3 words)

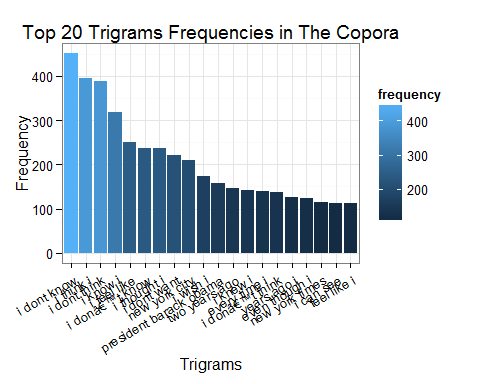
#### Trigrams Word Cloud

The word cloud shows the most common trigrams larger in scale.



#### Trigrams Histogram

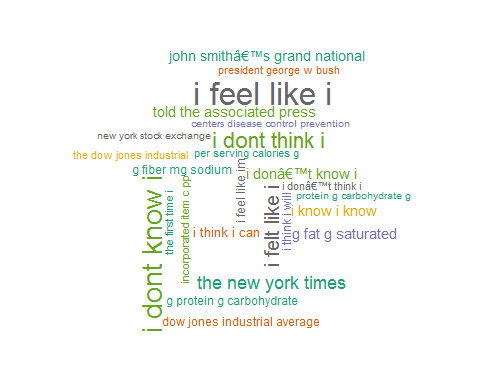
The histogram shows the different frequencies of the top trigrams.



### Most Frequent Quadgrams (4 words)

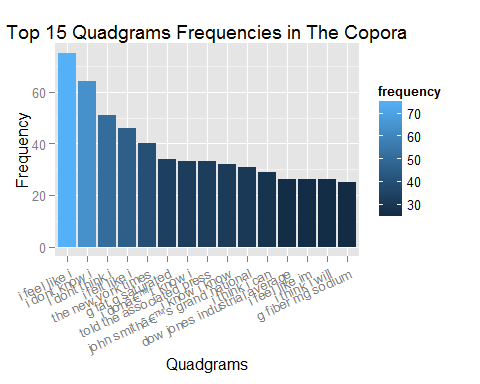
#### Quadgrams Word Cloud

The word cloud shows the most common quadgrams larger in scale.



#### Quadgrams Histogram

The histogram shows the different frequencies of the top quadgrams.



# Shiny App

## Text Prediction Strategy

* Preprocessing a specific language corpora using the HC copora or augmenting it with other corpora.
* Building ngram models from 1 and upto maybe 5-7 words depending on the richness of the corpora.
* Building a table for the frequencies of each different ngram model.
* Merging the different tables into one consolidated frequency table for all the ngram models.
* Given a text string of n words, the prediction algorithm will match these words in any of the ngram tokens in the frequencies table, starting with the n+1 gram tokens, if it does not match any n+1 gram token, the last n-1 words of the input string is matched in the n gram tokens this time and the process continues until a match is found or the last word, i.e. 1 word of the input text is matched in one of unigram tokens.
* If input is matched, the last word of the n+1 gram tokens with the highest frequencies are recommended for the user to complete his/her input text.

## Shiny App Plan

* Deciding on the optional cleaning steps to include in the final prediction model. For example, will the final model include stopwords and profanity words?
* Looking for augmentary english corpora that can be used to enhance the prediction to be built.
* Completing the n-gram model, I have already built the unigram, bigram, trigram, and quadgram models. However, I might be considering ngram models based on 5-7 words as well if the dataset provided is rich enough to enable me to extract these models.
* Deisgning and prototyping the user interface for the shiny app.
* Implementing the Shiny App.
* Documenting the App.
* Developing a short pitch for the Shiny app.

# Conclusion

This document reports the main data features of the HC Copora downloaded from <http://www.corpora.heliohost.org/>. For each text file in the English copora, the report shows the size in addition to other summary statistics such as, wordcount, number of lines, and the mean number of words per line. After building the unigram, bigram, trigram, and quadgram models, the report shows the most frequent n-gram words/tokens along with their frequency in the English Corpora. Finally, the report elaborates the text prediction strategy and the plan for the development of the Shiny App, the final product.