Building predictive model(s) for word prediction: Preliminary Analysis Report

[Author: Prashant Ratnaparkhi]

Executive Summary:

The goal is to build a predictive text model to predict next word in a sequence of words. This preliminary analysis of the raw text data is performed with the stated goal in mind. The raw data consists of three files containing twitter messages, news articles and blogs written in colloquial text. This report has 3 sections. First section summarizes the raw data and second section summarizes the results of cleaning & preprocessing the data.The third section contains important observations, remarks and next steps to achieve the goal.

1. Raw data Summary

The number of lines, words and word types (unique words) are tabulated below. This includes numbers, non-ascii text, symbols, swear words, mis-spelt words - essentially anything & everything.

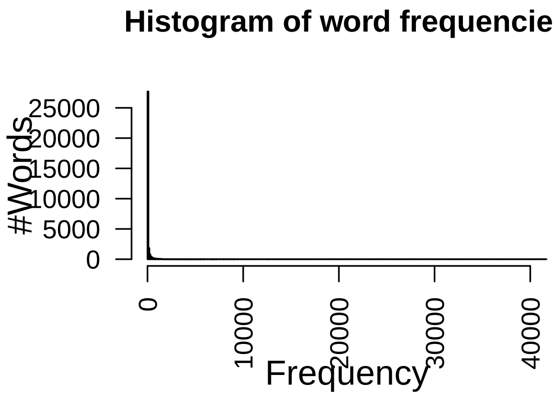
```
##            fileName numLines numWords numWordTypes
## 1 en_US.twitter.txt  2360148 31105049       336039
## 2    en_US.news.txt    77259  2755778        84974
## 3   en_US.blogs.txt   899288 38601569       309279
```

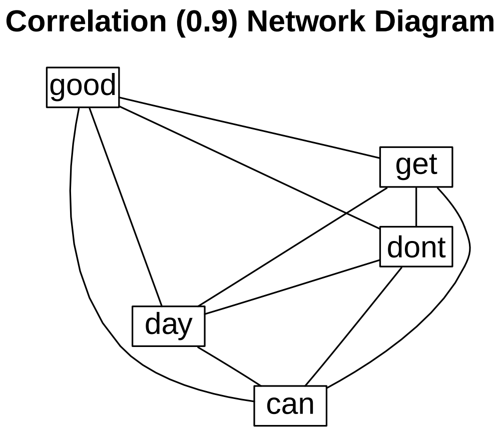
2. Cleaning, pre-processing & corpus creation

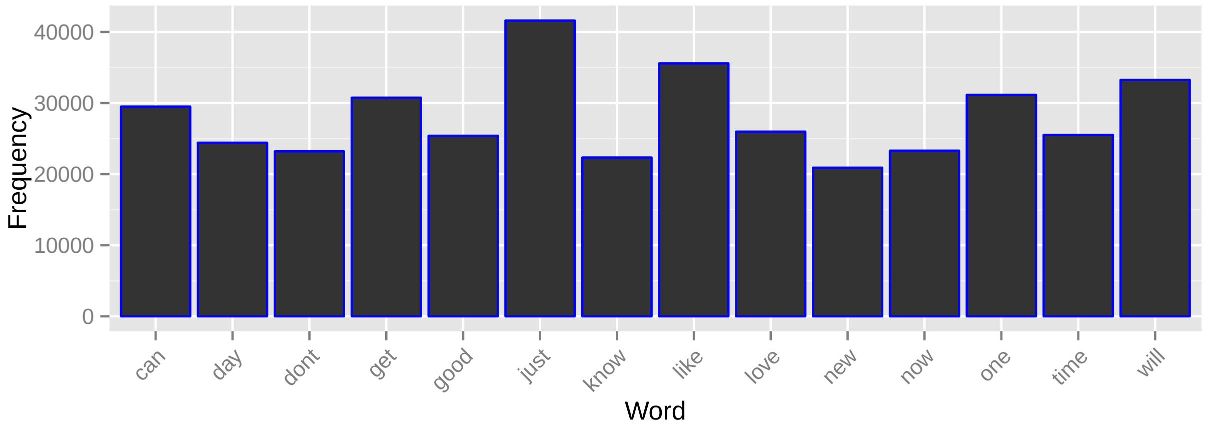
Due to processing time constraints, a random sample of 20% of raw data is selected to create a corpus & term document matrix using the 'tm' package. Cleaning & preprocessing involved removal of: (i) non-ascii characters, (ii) swear words, (iii) numbers, (iv) punctuation marks, (v) mis-spelt words, (vi) English stopwords (words such as 'the', 'am'etc.) Summary of term (word) frequency distribution is shown below:

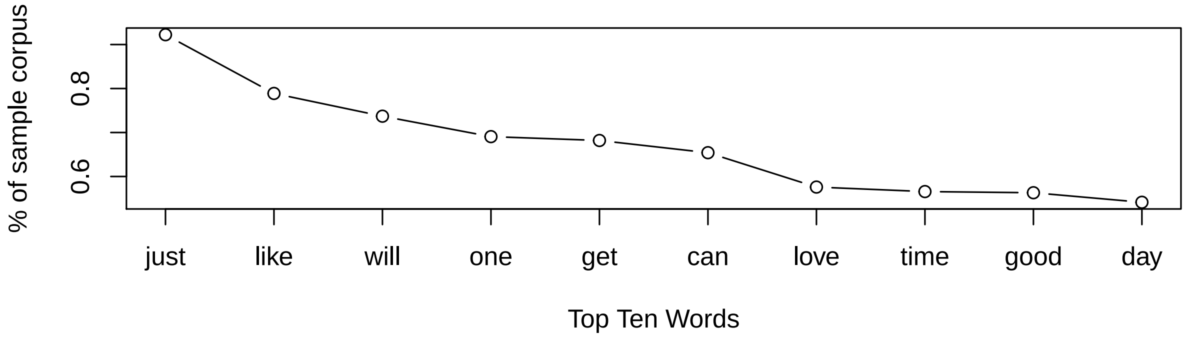
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##       1       2       9     138      42   41600
```

In the sample: (i) 22 terms are used with average frequency, (ii) 5375 terms are used with minimum frequency, and (iii) 1 term(s) are used with maximum frequecy. Most terms (or words) are used only once. Word frequency histogram and word associations for top five words are shown in the panel below.









The first plot above is a bar plot showing words with frequency of usage greater than 20000 and the second plot shows percentage of usage for top 10 words.

3. Observations & next steps

(1) The most challenging part, so far, has been the size of the raw data. This R code with 20% sample data takes approx 2 hours on 64 bit/8Gb/4 CPU Windows server. With 2% sample data it takes approx. 10 minutes. Most time is spent in creating corpus and term document matrix.

(2) The term document matrix built in this analysis, forms the uni-gram representation of the sample data set, and can be used for feature selection in subsequent steps.

(3) The next steps include: (i) Selecting & training an N-Gram language model. Choices include tri-gram models based on linear interpolation, Katz backoff (with appropriate smoothing techniques) etc. (ii) Rapid, iterative evaluation (using cross validation and test samples) of the models and revisiting step-(i), as needed to finalize a particualr model. (iii) Scaling of steps-(i) and (ii) to handle larger sample sizes - ideally 60% of raw data for training, 20% for cross validation and 20% for testing.