The milestone report

Miao YU

2014/09/17

Introduction

Portable office actually means the works done on the cellphone and the tablet and we need input system to saving our time on typing on them. So a smart and efficient keyboard is required and the core of this input system is a predictive text model. This milestone report is focused on this model, covering the very beginning, namely data collection, to exploratory analysis of the data set.

Data Collection

The data were downloaded from the course website (from HC Corpora) and unzipped to extract the English database as a corpus. Three text documents from the twitter, blog and news were found with each line standing for a message.

Data Pre-Summary

After scan the three documents with `bash`, I found the following features:

- the basic summary of the data set is shown as follows:

	line counts	word counts	document size
twitter	2360148	30373603	166816544
news	1010242	34372530	205243643
blogs	899288	37334147	208623081

Table 1: Summary of the datasets

- twitter is short(of course less than 140) with a lot of informal characters and less grammar, which means more noise
- news is written in a formal manner but the topics is focused
- blog's style is between the twitter and news with less noise and more topics
- the average length of each lines in the three database: blog > news > twitter, which means blog is the longest document class and longer document will help to build a better model for prediction in certain context

So, the blog data will be good for us to build a model if those three document is too large to be loaded for exploring. However, using sampling will ease the burden on the calculation and finally I sampled 30,000 20,000 and 10,000 lines with seed from the blogs, news and twitter database for exploring and training a model and the left data will be sampled to make the test data sets.

Tokenization

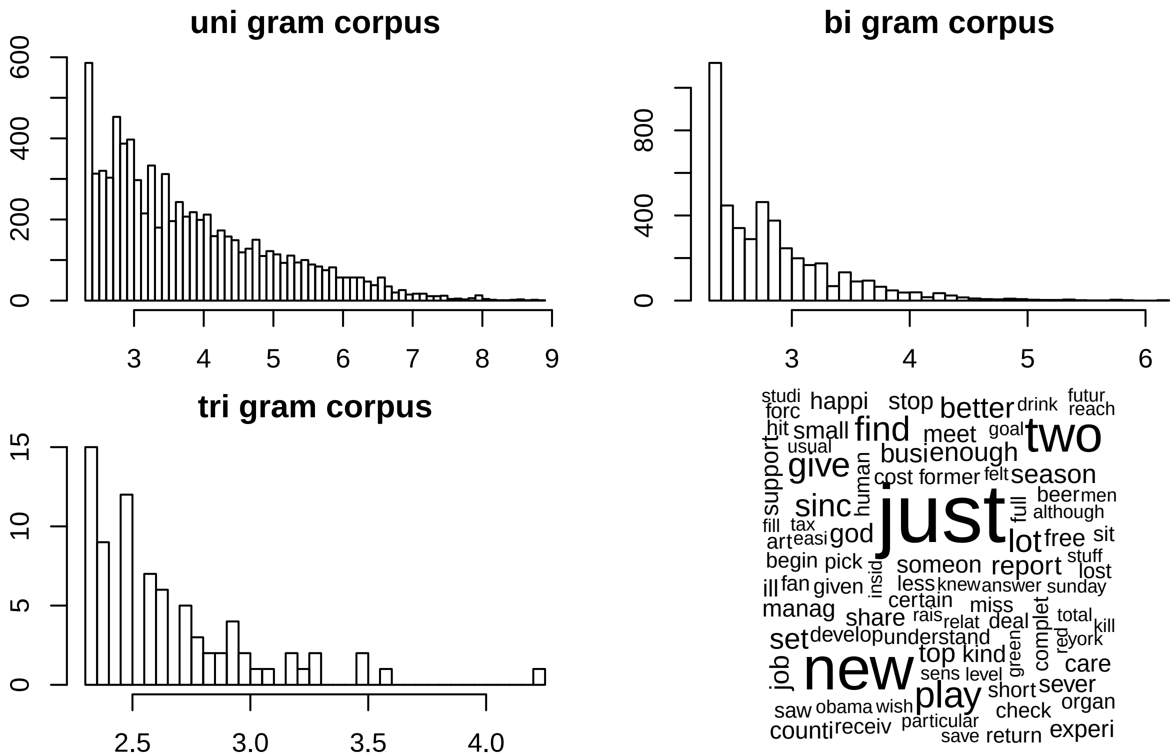
The whole tokenization is aiming at removing meaningless characters and the words with low frequency in the corpus. The final corpus will show the words or n-gram with a high frequency which will be helpful for exploring the relationship between the words and building a manful statistical model.

So, I extracted 1)the ASCII characters, 2)change the capital characters to lower case, 3)remove the punctuation, 4)numbers and 5)stop words and 6)stemming the left words. To decrease the spares of the term frequency, I removed the terms occurred less than ten times in the whole document to get the final corpus.

Exploratory analysis

To build a n-gram model, I extracted n-gram corpus with the help of `RWeka` package. The uni gram terms corpus has 8063 words, the bi gram corpus has 4551 terms and the tri gram corpus has 78 terms. Then I explored three corpus(uni gram, bi gram and tri gram) and made a histogram to show the distribution of the terms in them.

Figure 1: Histogram of term frequency and word cloud of all of the three corpus



As shown in Figure 1, the logged frequencies in all of the three corpus were still skewed to the left, which mean the sparse of the terms data. So I think it will be hard to build a good generation regression model but local regression would be OK. Also I found only 8063 words occurred more than ten times in the sampled documents compared with nearly 70 thousand words in an online dictionary, which mean focused on little words would work in most of the prediction. The word cloud showed the terms occurred more than 400 and those terms would be good to build a classification filter models before using a n-gram model to speed up the whole prediction.

OK, the exploratory analysis has inspired some features about the final produce: **hierarchical local regression model**.