Jacob Dineen

Text Mining

8/5/18

Homework 3

Due to the open-ended nature of this assignment, I’ve decided to do a bit of exploring with the NLTK (Natural Language Toolkit) module in python. The prompt asks us to select an existing corpus, or create a new one. I have decided to utilize one of the many corpuses engrained in the NLTK module for exploratory analysis on linguistic styling. The data I have chosen to work with is labelled as the Brown Corpus, which was “was the first of the modern, computer readable, general corpora” [1]. Essentially, we have a corpus that contains 15 clusters of labelled genres, with each containing some distribution of greater than 1.1mm words. This seems a bit daunting for some of my early work with text cleaning, so I’ve decided to subset the original corpus into two categories: News and Fiction. This corpus is over 50 years old [1], but what I want to identify is if there are any semantic or linguistic differences between the two styles of writing. I’ll likely explore the vectorized matrices of each genre, and potentially venture into some feature weighting/ranking analysis to show me which types of words are most likely to characterize which genre. This is strictly exploratory in nature – I understand styles and language evolve over time, and what was once commonplace 50 years ago may not be generalizable to the current environment.

To read in this data, one simply needs to download the nltk module, and the brown dataset, as shown in appendix 1.1. The data appears to all be stored in one large dictionary, like a Json data structure, so it is relatively easy to extract what I’m looking for, which is the vocabulary. I split out the distinct categories because I want to be able to look at them individually. This results in two variables containing each of the respective vocabularies; The news corpus containing roughly 100k words, and the fiction corpus containing roughly 70k.

Next, I’d like to convert all the words to lowercase [Appendix 1.2]. I believe that each of these vocabularies are large enough that doing so wouldn’t impact a model’s ability to infer a class if that was the goal of this assignment. It will, however, bin some otherwise unique tokens into other groupings that might increase a feature’s information gain if using some form of term frequency vectorization. I’d also like to remove stopwords in an automated fashion – Luckily, nltk has a prebuilt list of stopwords, so this becomes a relatively easy task using list comprehension [Appendix 1.3]. In the case of inference, stop word removal would neglect giving more weight to features that appear often – So we wouldn’t have frequently-used words tilting our predictive abilities. I believe this would only really matter if we weren’t performing some vectorization with regularization, such as tf-idf, because common words are weighted lower than unique, less frequent words.

\*An interesting thing I went back and did here is compute the percentage of non-stop words to stopwords. The news corpus had 64% words that were not included in the stopword list, while the fiction corpus had 57% of words that were not included in the stopword list (App. 1.5).

I also went back and toyed around with some of the functions defined in the NLTK docs. One being lexical diversity, which takes the vocabulary size divided by the total word count and returns a diversity score. This part I found a bit interesting, as the news corpus had a higher diversity score than the fiction score. One might think that works of fiction might contain more unique words and linguistic styling, but in the case of these sets of documents, that was not the case. Logically, it makes sense that journalists might use more unique words to avoid reader fatigue and generate greater interest (App 1.6). This was done with stopwords already removed from each of the respective corpuses.

Another interesting thing, I thought, would be to look at the ratio of unusual words in each corpus. This is done by: “Filtering a Text: this program computes the vocabulary of a text, then removes all items that occur in an existing wordlist, leaving just the uncommon or mis-spelt words” [2]. The percentage of unusual words was slightly higher in the fiction corpus than the news corpus – This makes sense because works of fiction may take freedoms with the English vocabulary, particularly when world-building in an creative environment (App. 1.7).

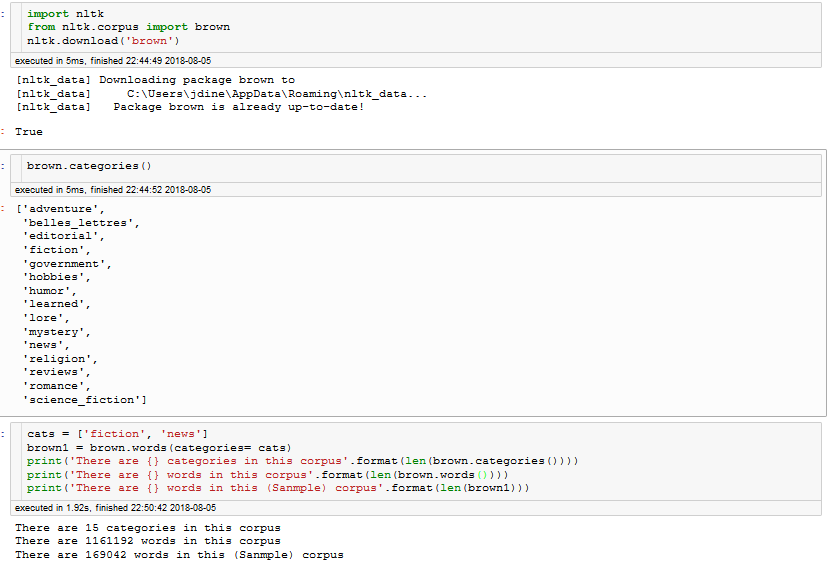
Looking at the most frequently-used words in each corpus is also a relatively easy task – NLTK has a built-in counter function to look at the frequency distribution of a passed list, and a function to sort and extract the most common n features. This analysis isn’t as enlightening as I had thought it may be, as shown in Appendix 1.4. Really the only thing I can see at a surface level looking at the top occurring words is that the News corpus does appear to have a more declarative slant, as noted by some of the top entities recognized. The fiction corpus also used exclamation points a lot more than the news corpus – This was a mistake on my part by not removing punctuation first, but maybe the tokenized punctuation could be telling of which genre a document belongs to- One would venture that news articles are less likely to contain emotion as much as works of fiction.

All in all, I have done some exploratory work on each of the sampled corpuses to understand any similarities or differences between the two genres. To the human eye, they are stylized in a comparable manner, and without prior knowledge, I likely wouldn’t be able to tell the difference between the two. Things that could have been done to improve analysis:

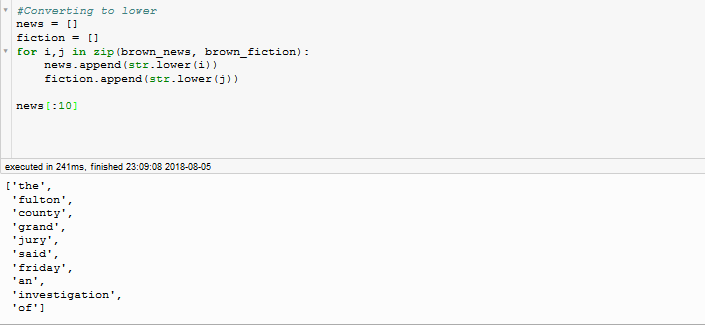
TF-IDF vectorization with labeled sets. Run these through a Bayesian model or a random forest and compute entropy+information gain to see which words are mapping a doc to a class. I assume this will be done on future homework assignments.

**Appendix**

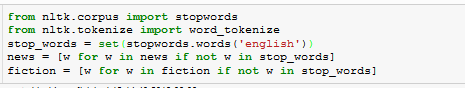
**1.1 – Data read in**



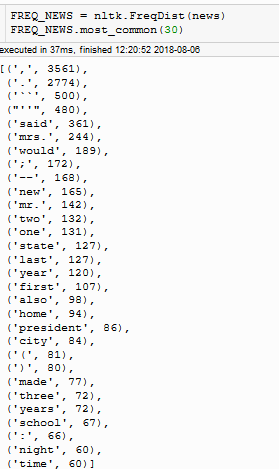
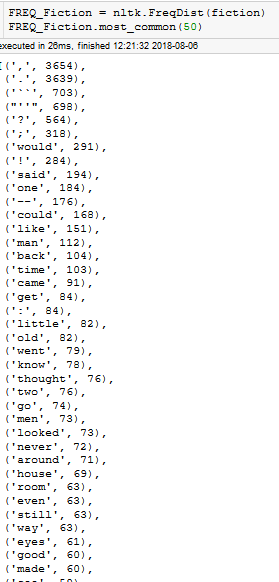
**1.2 – Converting to lowercase**



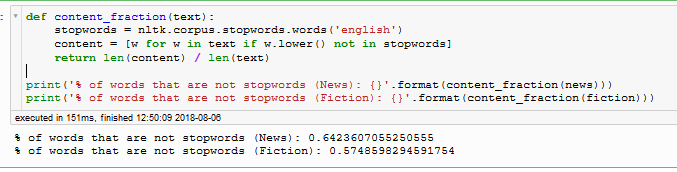
**1.3 – Removing Stopwords**



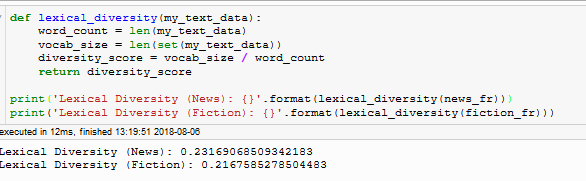
**1.4 – Frequency Distributions**



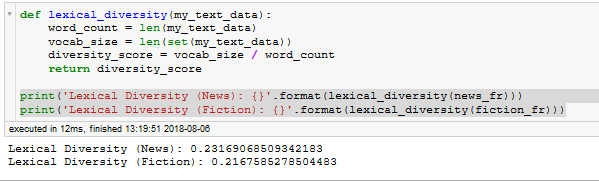
**1.5 – Distribution of Words to Stopwords**



**1.6 – Diversity Score**



**1.7 – Unusual Word Ratio**



**References**

[1] <https://www1.essex.ac.uk/linguistics/external/clmt/w3c/corpus_ling/content/corpora/list/private/brown/brown.html>

[2]

<https://www.nltk.org/book/ch02.html>

[3]

<http://scikit-learn.org/stable/modules/feature_extraction.html>