COMPSCI 4NL3 Homework 1: Counting Tokens

Alan Zhou

January 2025

1. **Data**

   The file chosen for this Homework comes from the Project Gutenberg source. It contains the contents of a book called "From the West to the West: Across the Plains to Oregon" by Abigail Scott Duniway.

   I originally selected this book at random, however found the potential in this book after reading over the preface. Specifically, the words "illiterate, inexperienced settler" caught my eyes. From this, I believe the rest of the book will describe a personal journey of self-education and creativity, an inspiring concept.

2. **Methodology**

   The software performs text normalization on plain text files. It analyzes the frequency of words and then generates a plot of token frequencies. The text normalization options are as follows:
   Lowercase: The lowercase option converts all tokens to lowercase to make sure all words, capitalized or not, are treated the same.
   Stemming: The Stemming option applies the PorterStemmer method from package nltk to reduce words to their root forms.
   Lemmatization: The Lemmatization option applies the to reduce words to their dictionary form. Lemmatization takes into account the word's meaning and context.
   Remove Stopwords: The Remove Stopwords option filters out common words that do not provide much value and occur frequently. The list of stopwords featuring "the", "and", "to", "of", etc... were generated through asking ChatGPT.
   Custom Option: The Custom Option is an additional option I introduced. It removes any token that is only one or two letters long. This can be useful when attempting to identify more meaningful words when analyzing. After reading some of the text, I find words such as "I", "a", "to", etc...show up often, but do not have much value in token frequency analysis.

   After the preprocessing, the software counts the amount of instances for each unique tokens. This is achieved by the function count_tokens, allowing for a clear understanding of the most frequent words in the dataset.

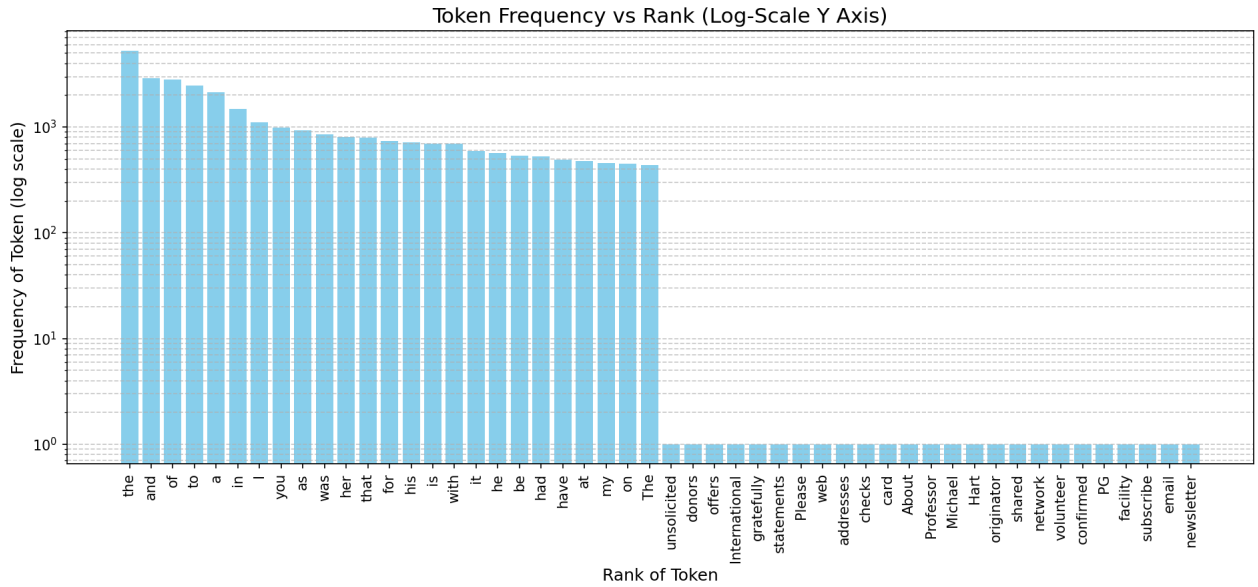
   To visualize, the function plot_token_counts displays a bar chart showing the frequency of the tokens using a log-scale y-axis by using libraries matplotlib and numpy. It shows the top 25 and bottom 25 token counts, providing an understandable view of the distribution of token frequencies.

   For the user to interact with the software, the user is able to enter "python normalize_text.py myfile.txt with a combination of arguments (<your-options>)" from the command line. This software can be easily manipulated for different datasets and varying cases.

3. **Sample Output**

   Given the command line script: python normalize_text.py textfile.txt lowercase:

   The visual bar graph output is:



The exact count of the top 25 most frequent and bottom 25 least frequent words are shown below:

```
Top 25 Most Frequent Tokens:
[('the', 5287), ('and', 2916), ('of', 2831), ('to', 2475), ('a', 2129), ('in', 1473), ('I', 1102), ('you', 993), ('
as', 927), ('was', 859), ('her', 809), ('that', 792), ('for', 733), ('his', 718), ('is', 692), ('with', 692), ('it'
, 590), ('he', 571), ('be', 536), ('had', 530), ('have', 490), ('at', 474), ('my', 459), ('on', 448), ('The', 440)]

Bottom 25 Least Frequent Tokens:
[('unsolicited', 1), ('donors', 1), ('offers', 1), ('International', 1), ('gratefully', 1), ('statements', 1), ('Pl
ease', 1), ('web', 1), ('addresses', 1), ('checks', 1), ('card', 1), ('About', 1), ('Professor', 1), ('Michael', 1)
, ('Hart', 1), ('originator', 1), ('shared', 1), ('network', 1), ('volunteer', 1), ('confirmed', 1), ('PG', 1), ('f
acility', 1), ('subscribe', 1), ('email', 1), ('newsletter', 1)]
```

4. **Discussion**

Observing the output of the top 25 most frequent words: they all seem to be functional words in the english language. These are articles, pronouns, prepositions, conjunctions, and other types of words required for correct grammar in sentences.

On the other hand, the 25 least frequent words seem oddly specific, all with a total frequency of 1. I assume that these words are not a part of the story, but rather apart of legal documentation for the file or other information regarding specifics such as sources, websites, etc... In addition, I believe that the words "Professor", "Michael", and "Scott", must be a proper noun.

The wikipedia page is regarding Zipf's law, a formula describing word frequency in natural language. Quoting from the subsection *Word frequencies in natural languages*: "In many texts in human languages, word frequencies approximately follow a Zipf distribution with exponent s close to 1; that is, the most common word occurs about n times the n-th most common one." In the above described case: this law does not apply very well. This is likely due to the fact that so many of these functional words and stopwords are present, making the ratio higher than Zipf's distribution would describe. The very bottom (the low frequency words) are likely to follow Zipf's distribution.

The impact of removing stopwords like "the" are significant, since they occur so frequently, regardless of the subject matter. Compared to removing regular content words, there is much less loss of meaningful content when removing these stopwords.

Unexpected issues I ran into includes the processing of the file, which I quickly solved through research on the Internet. I learned how to use some techniques from the packages nltk, for lemmatization, and well as stemming. I previously knew from general knowledge that common words like "the" are extremely common, but had not known to what extent, which I learned after this counting tokens homework.