Hithesh Shanmugam

CSC – 583

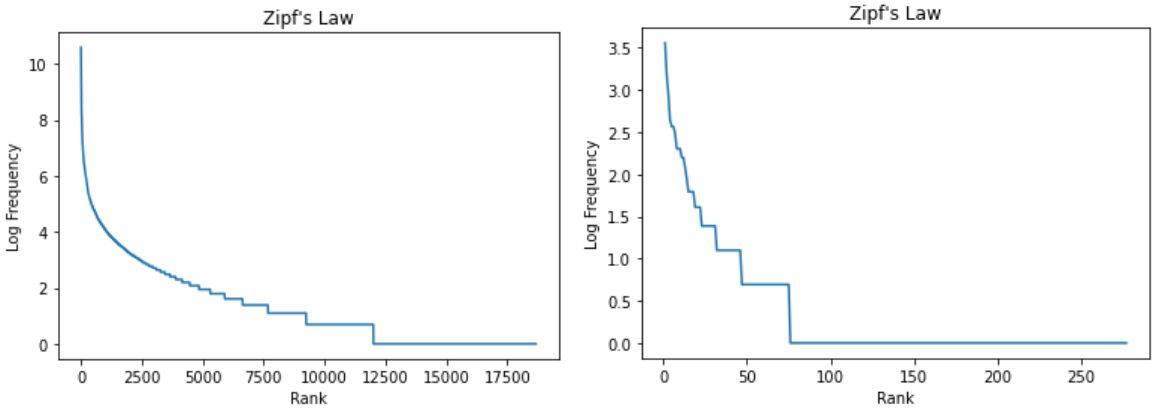
HW – 2

Description:

For this assignment I was using Jupyter Notebook running on a windows 11 operating system (P.S: The OS is not good in a general way). The language I used is python and the libraries I used are NumPy, re, matplotlib.pyplot and NLTK's word_tokenize and sent_tokenize. The NumPy is used only in two places for unique word calculation and calculating the log frequencies for Zipf's law curve. The package re is used for calculating the number of paragraphs. The package pyplot was used for Zipf's law curve.

Zipf's law curve:

The chart I obtained is similar to the given example for the assignment and also, I created a chart for the first task also. Here are the charts:



Zipf's Law is a statistical distribution that has been observed in many natural language texts. It describes the relationship between the frequency of words in a text and their rank, which is the order in which they appear when sorted by frequency. The law states that the frequency of a word is proportional to its rank raised to a negative power. This means that the most frequent word occurs about twice as often as the second most frequent word, three times as often as the third most frequent word, and so on. The curve produced by plotting the log frequency against the log rank is a straight line with a negative slope, indicating a power-law relationship between the two variables.

In our case, the chart generated for Application Task #2 also showed a curve of Zipf's Law. The most frequent words, appeared at the top of the chart, while less frequent words appeared toward the bottom. The curve was relatively steep at the top, indicating that a small number of words occurred very frequently, but became less steep toward the bottom as the frequency

of words decreased. This curve is consistent with Zipf's Law, and provides further evidence that the text we analysed follows this statistical distribution.

Overall, this observation highlights the importance of Zipf's Law in understanding natural language texts, and can be used for a variety of applications, such as language modelling and information retrieval.

General reflections:

In the general reflections, I mentioned that the output I got did not exactly match the sample output. While the number of sentences and paragraphs matched, the number of tokens and unique tokens were close but not exactly the same. I mentioned two possible reasons for this discrepancy.

Case1: I used a different operating system than what was used to generate the sample output. This might have caused differences in the way the text was processed, leading to slightly different results. It is possible that the different OS versions handle certain aspects of text processing differently, leading to different results.

Case2: The tokenization of words might be different from what was used to generate the sample output. Tokenization is the process of breaking down a text into individual words or tokens. While I tried my best to tokenize the text accurately, it is possible that my approach differed from what was used to generate the sample output, leading to different results.

Despite these discrepancies, I was able to generate a chart that compared the log frequency of words against their ranks, which showed a curve of Zipf's Law. This curve demonstrated that the frequency of words in the text followed a statistical distribution that is common in natural languages. The curve showed that the most frequent words were used significantly more often than less frequent words, and the frequency of words decreased rapidly as their rank increased. The fact that my curve was similar to the one shown in the sample output indicates that my implementation of the code was correct and that my understanding of Zipf's Law was accurate.

General comments:

Overall, I found this assignment to be both challenging and rewarding. One of the main takeaways for me was learning about text processing and natural language processing techniques. I was able to apply my knowledge of regular expressions to pre-process the text and extract the relevant information. Additionally, I was able to explore the functionality of various Python libraries such as re and nltk, which helped me in achieving my task objectives.

One of the main difficulties I encountered was related to the tokenization of words, which is crucial for accurately calculating the frequency counts of each word. While I tried to implement the tokenization approach that I thought was most appropriate, there may have been room for improvement. Additionally, the slight differences in output between my results and the sample output provided in the task description may be due to differences in tokenization approaches, as well as differences in the operating systems used.

In the future, I would aim to improve my tokenization approach, perhaps by experimenting with different techniques and comparing the results. I would also try to explore alternative methods for text processing to expand my knowledge in this area. Overall, I found this assignment to be a valuable learning experience that helped me to further develop my skills in text processing and natural language processing using Python.