**Natural Language Processing, CSCE-4290/5290**

In class task 9/2/2021 due at 11:59 PM

Please follow the following tutorial and solve the code for the expected program. After coding is complete answer the questions at the end of the document. Upload your completed task to Canvas on the ICE-1 tab.

Good Luck with your first NLP task 😊

**Tutorial**

**Tokenize text using pure Python**

First, we will grab a web page content then we will analyze the text to see what the page is about. We will use the [urllib module](https://likegeeks.com/python-programming-basics/" \l "Web-Crawling) to crawl the web page:



As you can see from the printed output, the result contains a lot of HTML tags that need to be cleaned. We can use BeautifulSoup to clean the grabbed text like this:



Now we have a clean text from the crawled web page.

Awesome, right? Finally, let’s convert that text into tokens by splitting the text like this:

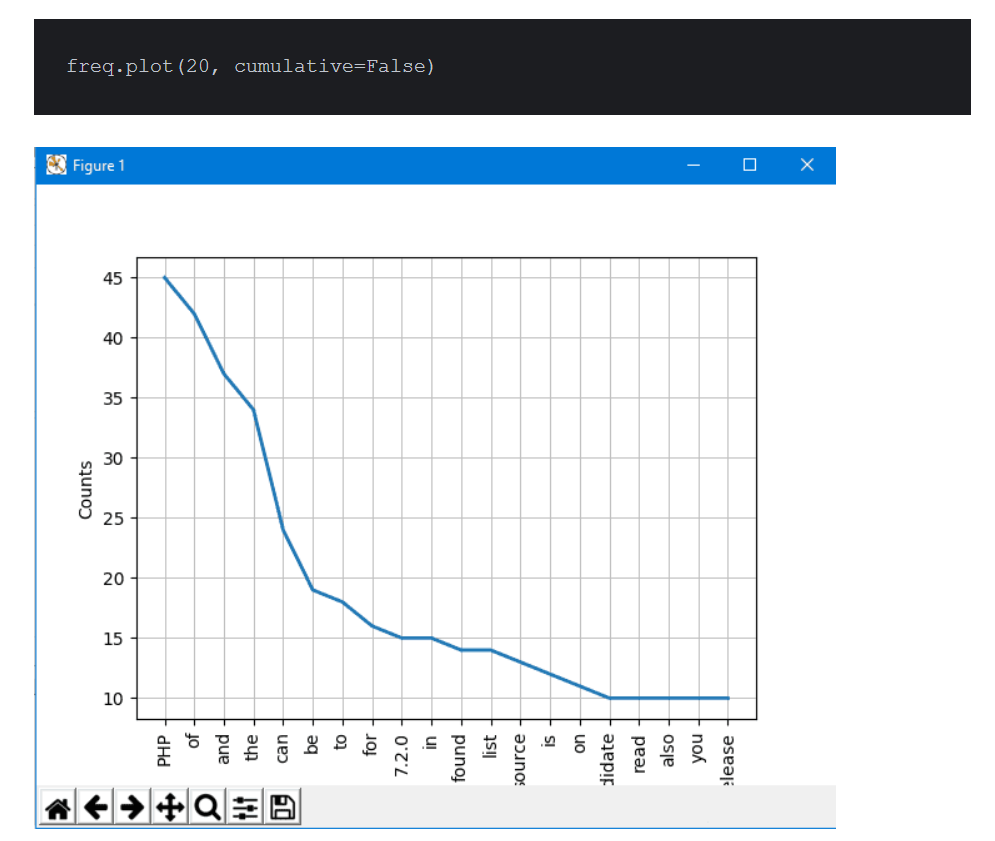


## Count word frequency

The text is much better now. Let’s calculate the frequency distribution of those tokens using Python NLTK. There is a function in NLTK called FreqDist() does the job:



If you search the output, you’ll find that the most frequent token is PHP. You can plot a graph for those tokens using plot function like this:

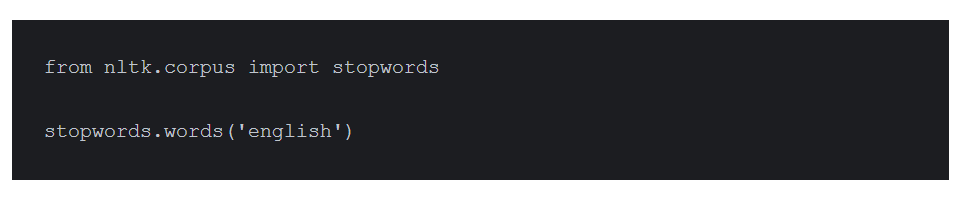


From the graph, you can be sure that this article is talking about PHP. Great!!

There are some words like The, Of, a, an, and so on. These words are stop words. Generally, you should remove stop words to prevent them from affecting our results.

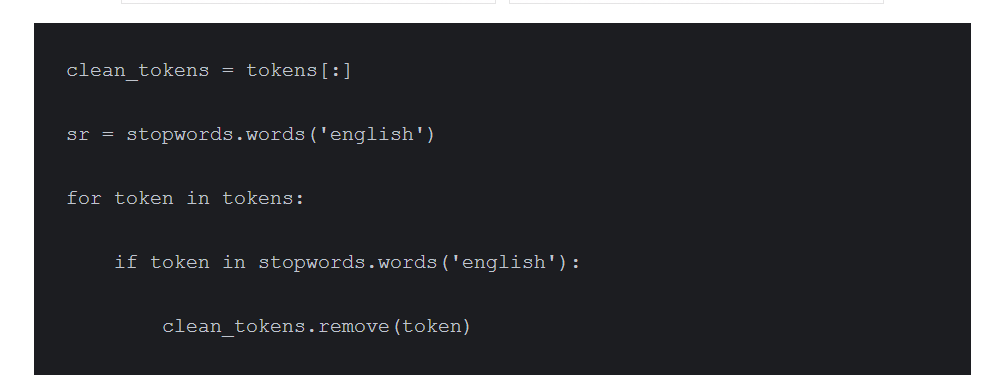
## Remove stop words using NLTK

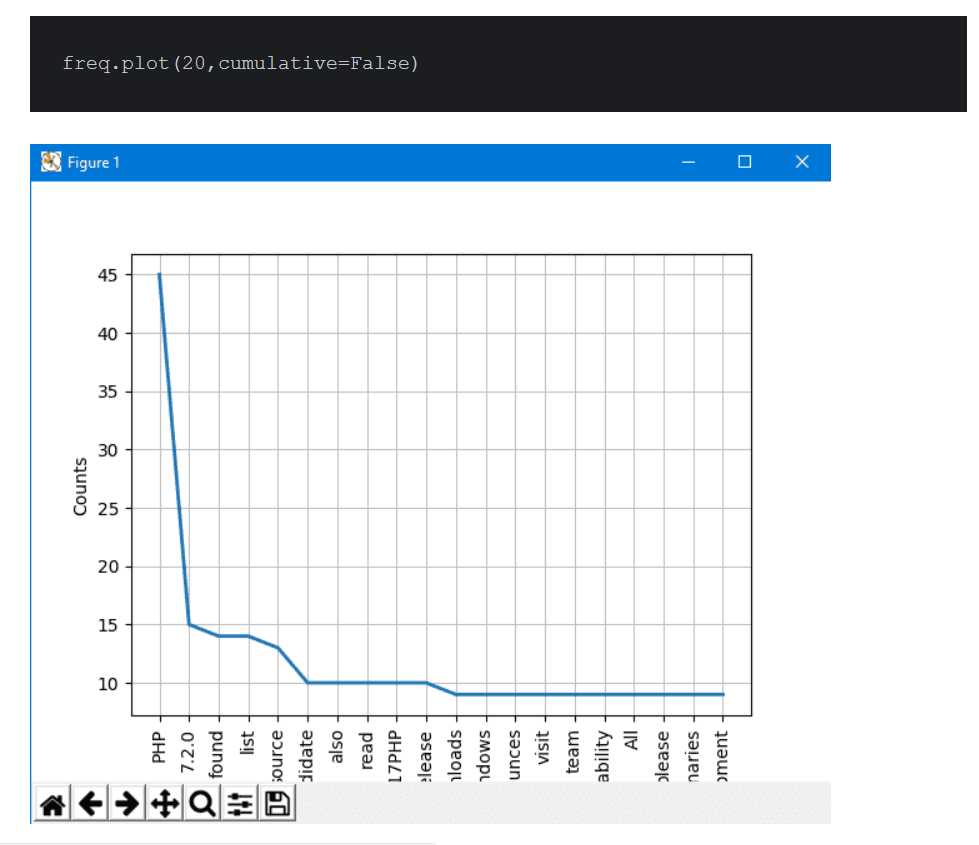
NLTK comes with stop words lists for most languages. To get English stop words, you can use this code:



Now, let’s modify our code and clean the tokens before plotting the graph.

First, we will make a copy of the list; then we will iterate over the tokens and remove the stop words:





**Coding Exercise:** Write a code to tokenize the text and grab contents from a webpage where you can find information about SpaceX. Use stopwords and use the above strategy to calculate the frequency. The graph should display the first 10 high distribution words in the webpage while ignoring others. If the frequency of the word is less the 5 times ignore those words as well.

**Questions:**

1. Why we use stopwords? Why stopwords are not necessary for NLP frequency distribution.
2. Based on high frequency words what information you can extract from the graph?
3. Can you provide different visualization for frequency distribution? If yes, please perform. If no, why?

Please upload you code along with outputs to Canvas. The submission should include code file, and word file for explanation. Make sure to comment your code properly.