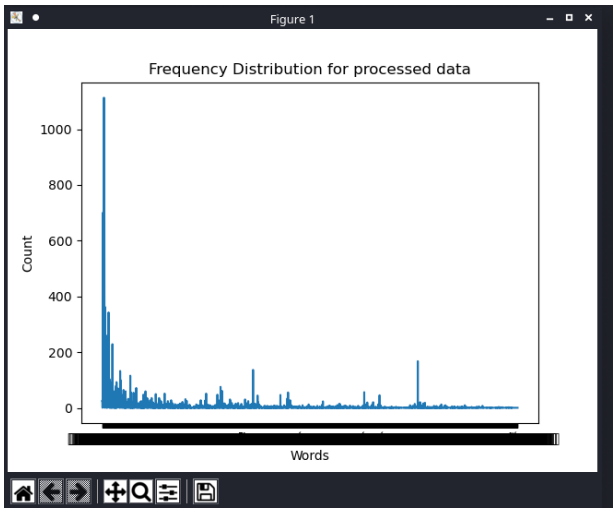
Computational Liniguistics - 1

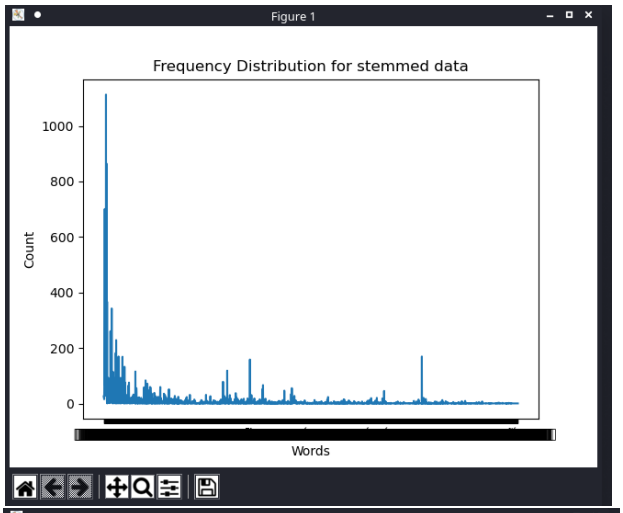
Project 1

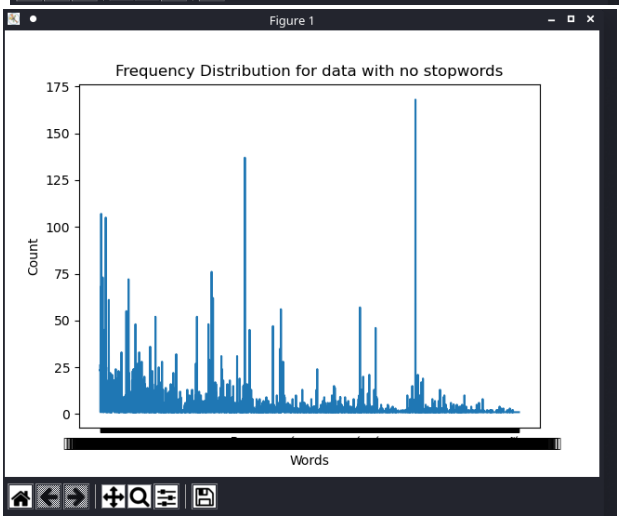
2020114012

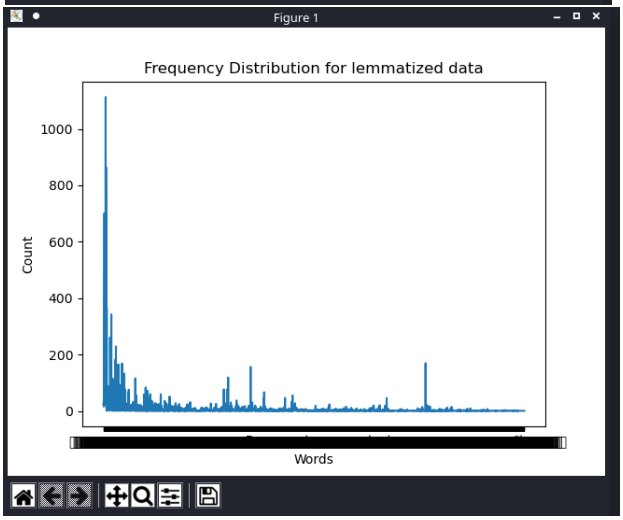
After running the word cloud generation code , I made 4 different frequency graphs for each of the languages , which came out to be as follows :

Hindi





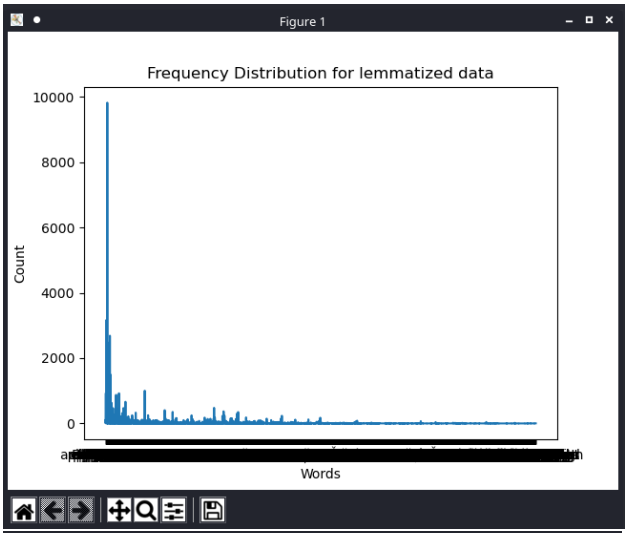


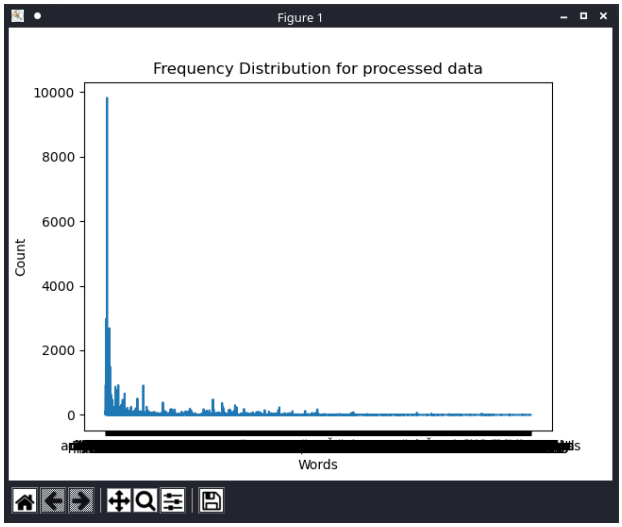


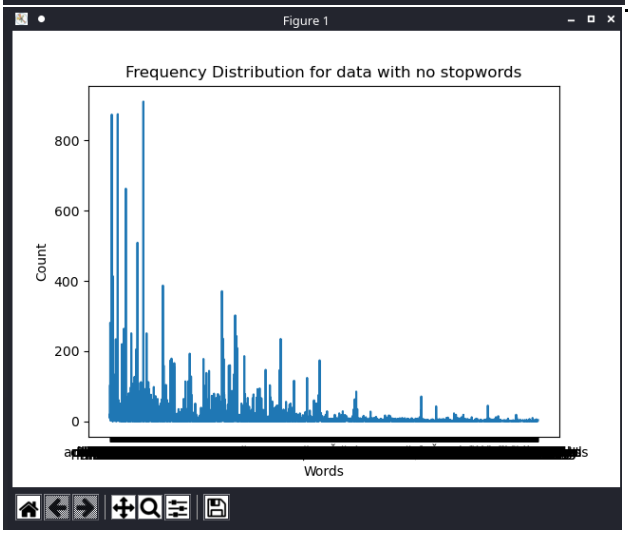
Analyzing these graphs I calculated the average occurances of the data with no stopwords , because we don't want any stopwords in our wordcloud , only around

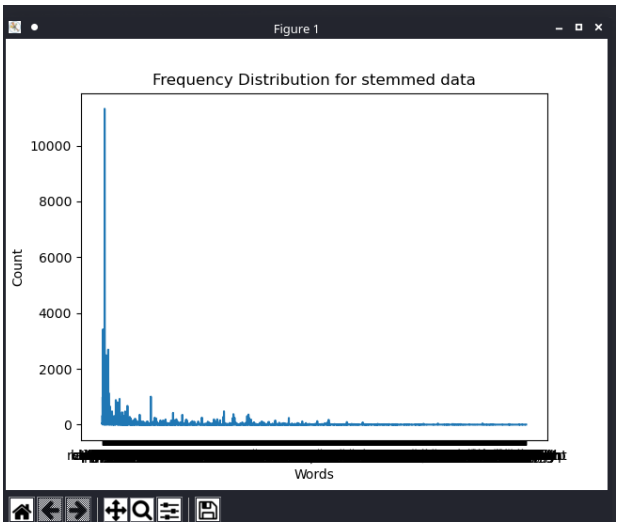
100 of them could cross the count of 10, hence I used this fact to make the wordcloud for about a 100 words , since the dataset taken was also fairly large.

English









The english dataset had a lot more variations in it , and frequency of the words was also amazingly higher than we saw in the Hindi text , the top 100 frequency words came about almost more than 100 times !

These graphs gave us an approximate idea on the number of words to be occuring in the wordcloud an dwould have an significant amount of contribution in our dataset.

Now the algorithm used to do so was as follows :

1. Crawl through specific URLs taken from Wikipedia and storing the content of all of them in a variable on which we will perform several operations.
2. The obtained content from the URLs was cleaned (removing symbols etc).
3. Now the clean data was sent into a tokenizer and a POS tagger courtesy of nltk.
4. The obtained token were then sent to a stopwords removal function which, as the name suggests , removed all the stopwords from the conent.
5. Now using the content without stopwords , we created a Frequency distribution and from that distribution we considered the 100 most occurring tokens , which we understood from the analysis of the graph.
6. These top 100 tokens were then considered to be added in the wordcloud which gave us the following 2 wordclouds for the 2 considered languages.