Project –1

Computational Linguistics – 1

Shubhankar Kamthankar

2020114004

Project Requirements -

1. Collect 10k sentences from both English and an Indian Language of your choice (Marathi in my case). [Done]
2. Use Crawling to extract the text. You can choose a particular website crawl. [Done]
3. Clean the corpus (Remove images, ads if any) [Done]
4. Remove foreign words/expressions, punctuations, symbols like currency, Abbreviations. Acronyms (WHO, UNICEF) can be retained. [Done]
5. Tokenization of the corpus. [Done]
6. POS tagging [Done]
7. Remove Stopwords [Done]
8. Stemming and Lemmatization [Done]

Tasks -

1. Frequency graphs for each of the above tasks. Analyse the Graphs.
2. Write your own algorithm to build the word cloud.
3. How many words do you want to include in the word cloud? Mention the reasons for your choice.

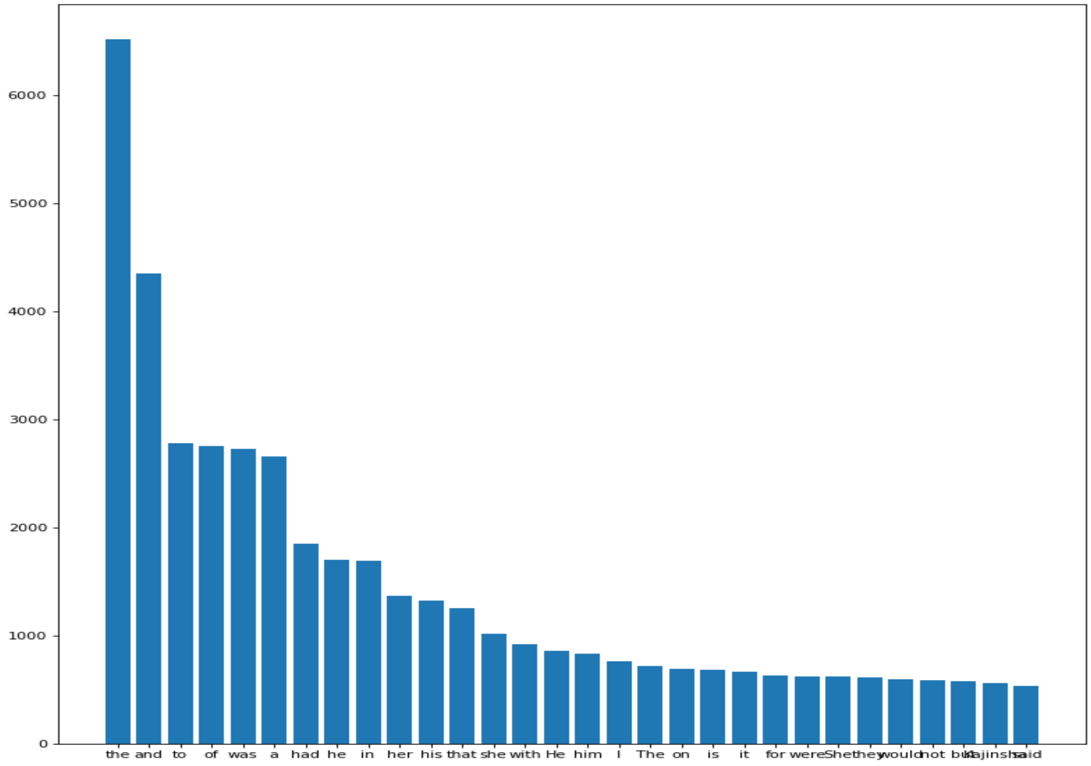
English

The Corpus is from a book “The Black Hill” by Mamang Dai, which is a historical fiction revolving around 3 main characters, set in Arunachal Pradesh. The corpus is roughly 10,000 lines long. For English Language, there being many precise (and abundant) resources, there were little to no problems faced while going over the tasks.

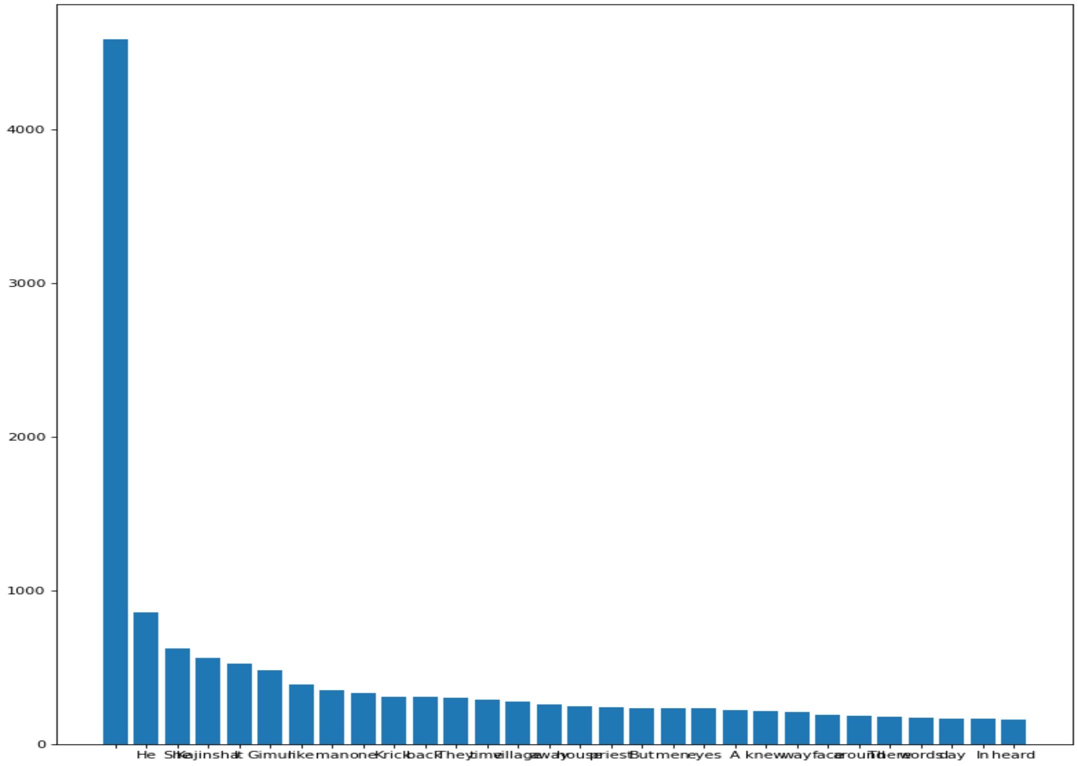
Marathi

The corpus is from a collection of books “The Sculptors of Maharashtra” which mentions achievements in the lives of few influential people in Maharashtra, throughout the past century. These influential people include Dadasaheb Phalke, Sane Guruji and Ahilyabai Holkar. The corpus is roughly 10.5 - 11k lines long. For Marathi, there was an acute availability of resources, and the ones available were not very accurate as well. For example, in the iNLTK package POS tagger, the adverbs and adjectives were tagged incorrectly in many places. The lemmatization tool was virtually ineffective as it showed no change in ANY of the words. Even upon searching very thoroughly, I could not come across any Marathi-based stemmer. The one I found was also written in JavaScript.

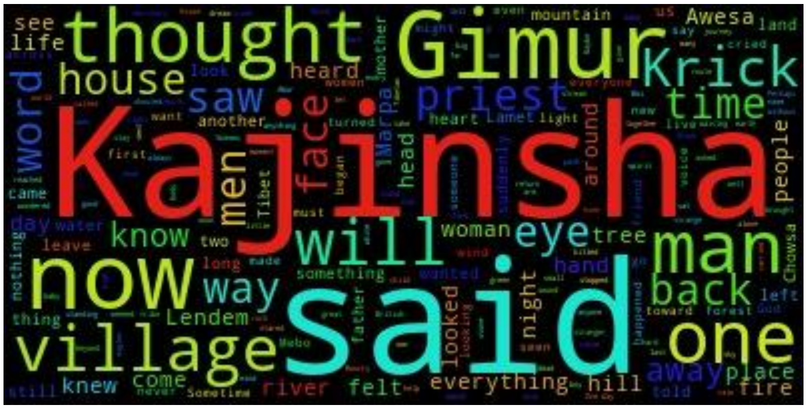
Frequency Graphs



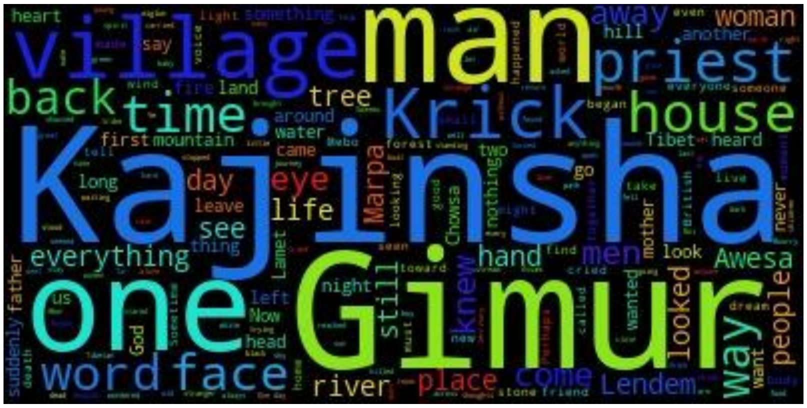
This graph shows the frequency distribution of words before the stopword removal is such that the frequency of the most frequent non stopword is comparable to one of the least occurring stopword. These are used much more often than non stopwords (named entities). On the adjoining graph, (next page, you can see the word distribution with the stop words removed. This graph appears to be much less skewed as compared to the previous one (with respect to absolute difference between the extremes).



The words like he, she have been retained as they as are context-dependent and hence carry important information. Below are the visualizations of the word clouds before and after removal of the stopwords.



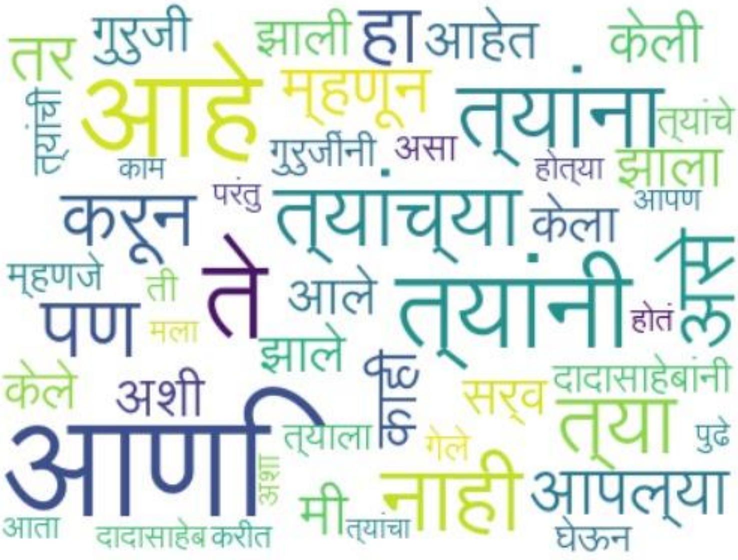
Before Removal of the stop words



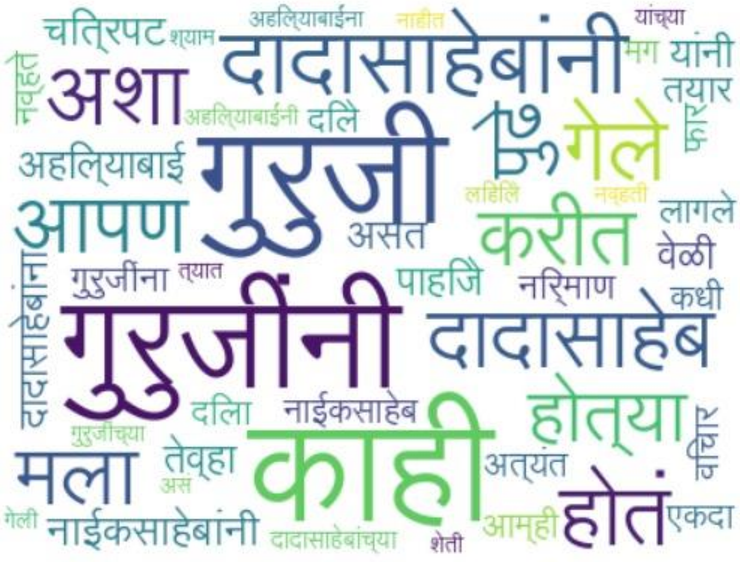
Post stopword removal

In English, the corpus contains a fairly balanced distribution of nouns (post-removal) and hence it made sense to take more words (50 in this case). For Marathi (as the word-cloud will show) the corpus had a much higher distribution of stopwords and verbs as compared to nouns or other non stopwords. Some errors encountered were inconsistent spelling of the words in the typed format, which was greatly distinct (wrong) as compared to the NLTK version.

The graphs were also not showing the Marathi text due to unavailability of the proper font [due to which I have made a.txt file for the frequency distribution].



Before stopwords' removal



After the removal of the stopwords.

Please find below the contents of the .zip file :

1. In each folder (Mar and Eng) :
    a. Freq.py - Python3 code to print a list of all the words indicating their respective frequencies
    b. Test.txt - Text file containing raw data for the respective languages.
    c. Sentence_counter.py - Python3 code to count the total words and stops ('.') in the document. The full-stop count is inaccurate as some part of the data has been initially cleaned manually.
    d. No_punc.py - Python3 code to remove punctuations from the corpus.
    e. Tokenlem.py - Python3 code to lemmatize the corpus.
    f. POS.py - Python3 code to tag the corpus according to Stanford POS Tagging rules.
    g. Word_cloud.py - Python3 code to generate a visual representation of the word cloud of the corpus.
    h. Bar_plot.py files – Python3 files to plot the bar graph of the most frequent words occurring in the corpus, before and after removal of the stopwords. One bar_plot file is missing in the Marathi folder as it would be redundant (font is not visible as distinct characters).
    i. .txt files with same/similar names as the .py files indicate the respective output of the programs stored in them.
    j. 2 .jpeg files denoting the visualization of the word cloud before and after removing the stopwords.

2. In English folder, stemmer.py and stemmer.txt respectively denote the code to generate the stemmed text and the text itself.
3. In the Marathi folder, Lohit-Devanagari.ttf is the font file for Devanagari script (in which Marathi is written)

In case the words the images or graphs aren’t clear enough, I have also uploaded them in the CL repository in GitHub.