**VORPAL SWORDS**

**Arnav Arora : 20ucs026**

**Aryan Singhai : 20ucs036**

**Anmol Jain : 20ucs022**

**Harsh Jain : 20ucs074**

# **Link to GitHub code repository**

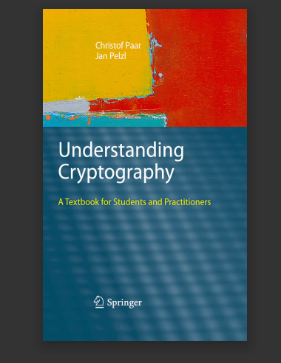
**https://github.com/thetazap/NLP\_Project\_Round\_1\_Vorpal\_Swords**

# **Problem Description**

To a book in .txt format and perform the following Natural Language processing operations on them

* Apply Data pre-processing on the text
* Generating frequency distributions of the words
* Creating word clouds from the text before and after removing stop words
* Evaluating relationship between word length and frequency
* Parts of Speech tagging for the words in the text

**Book chosen for applying the processing**



**Understanding Cryptography Foreword by Bart Preneel**

# **Python Libraries/Modules used**

Matplotlib : for drawing plots

Python re library (regular expressions library): For regular expressions

NumPy :for parameters of axes while plotting graphs

Nltk: :Used for tokenizing, removing stop words

Math :For calculating floor and ceil function values while plotting values

WordCloud :For creating word cloud

Collections :For getting the frequency mappings of the POS tags

Itertools : for iterating in dictionary

# **Inferences after examining raw data**

This raw text contains copyright related information, chapter headings, random blank lines and unprocessed text which cannot be directly processed.

# **Data Pre-processing and Preparation steps**

We performed the following data pre-processing steps

1. Removing chapter number and chapter Headings
2. Removing punctuation
3. Removing figures and tables
4. Removing fig. description
5. Removing tables description
6. Tokenising the text into a list of words
7. Removing chapter headings and unrelated data
8. Removing running section part

Text

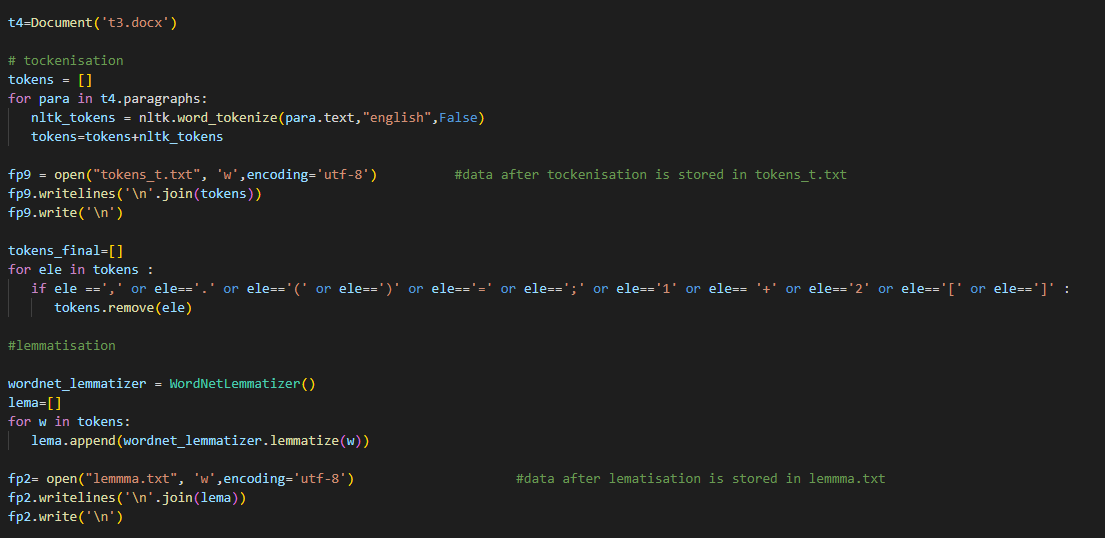
Description automatically generated

doc is the book in docx form which we have used for our project

in this step we have removed all the tables and stored the remaining in docx t1



We have opened t1 as t2 and by going through each paragraph in t2 we are removing all unwanted data ( Fig.1.2 , sect. 1.2 ,Table 1.2 etc.) and storing it in docx named doc\_1 . since images are not considered as Paragraphs so it will not be processed and will not be included in our doc\_1 file . and finally we have saved doc\_1 as t3 .



Now we have opened t3 as t4 and have done word tokenise and made a list of tokens naming tokens . and stored it in text file named tokens\_t . while analysing it we found some unwanted material (such as ‘=’ , ‘(‘ etc.) so we have removed those from tokens .

Then we have done lemmatisation on the tokens so that each word comes into its base form and stored it into list named lema and text file named lemma.

Text

Description automatically generated

After lemmatisation we have done steming , which completes our tokenisation process.

Now finally we have stemm.txt as our final tokenised text file and stem as final tokenised list .

**Frequency analysis**

Text

Description automatically generated

Text

Description automatically generated

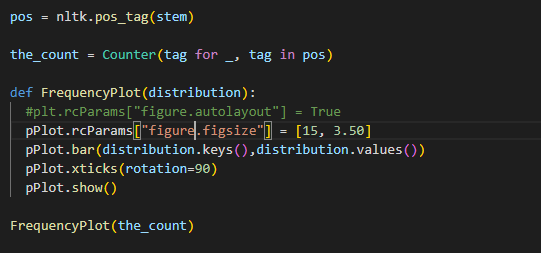
**Word Cloud**

A screenshot of a computer

Description automatically generated with medium confidence**Stop Words Removing**Text

Description automatically generated

**Tagging**

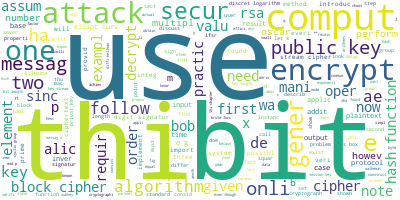


# **Illustrations (Word clouds and word wise frequency plots)**

**BEFORE REMOVING STOP WORDS**

Chart, histogram

Description automatically generated



**AFTER REMOVING STOP WORDS**

Chart, bar chart, histogram

Description automatically generated



# **Inference from word Clouds**

* The word clouds before and after removing stop words are quite different due to the high frequency of many of those stop words. One of the reasons may be that stop words can be used in a variety of contexts whereas nouns and verbs are more restricted to the situations to which they relate to.
* After removing stop words, we are able to find the set of words which provide us meaning and context about the document.

**Word length – frequency**

Here we are calculating the word length and their frequency of occurrence.

Chart, bar chart

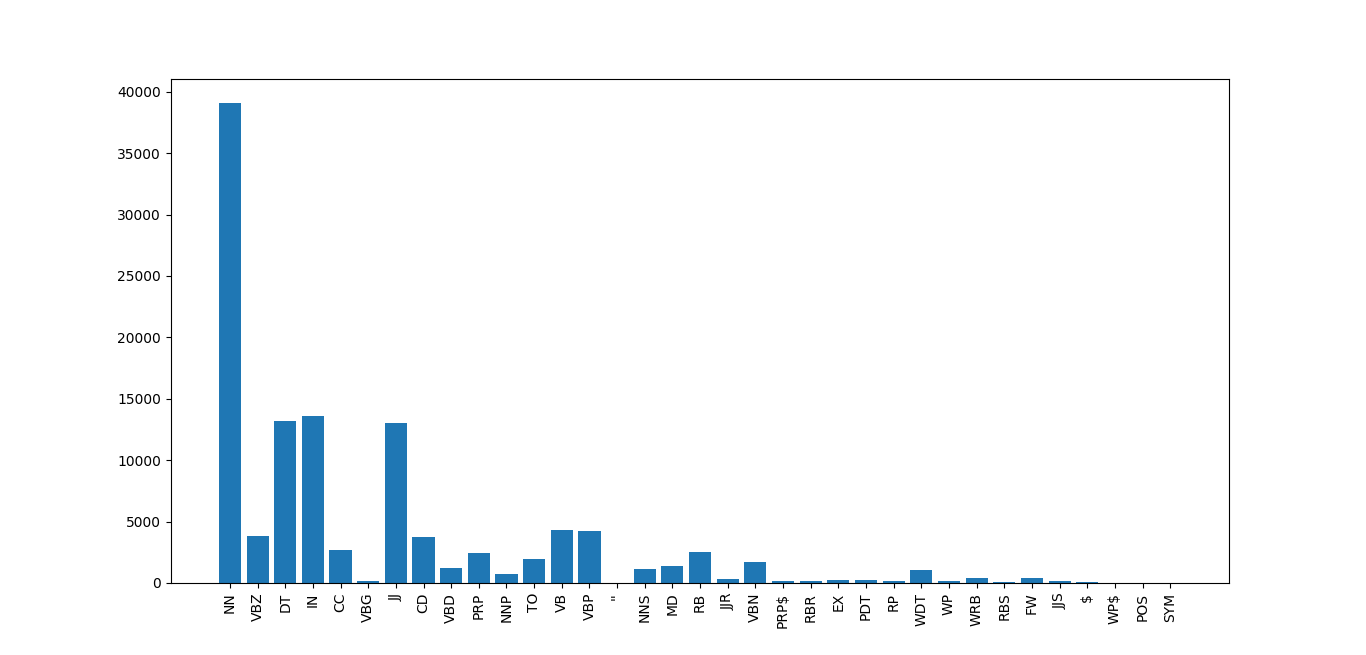
Description automatically generated

# **Inferences from word length- frequency plot**

* For this book Words having length between 1 to 3 are the most frequently occurring words in this book . After those words with smaller lengths (up to a certain length) are frequent followed by words of length 4 to 5. Very long words appear very rarely. Overall implying that most of the words lie in the length range of 1 to 3.

# **POS tagging**

Here we are finding the tag associated with each word that was pre-processed.



# **Inferences POS\_tagging**

We applied pos\_tagging on the two book using pos\_tag function. The pos\_tag(words) function uses the Penn treebank as the default tag set as per official documentation.

In our book the most frequently occurring POS Tag is ‘NN’ with count 39116 followed by ‘IN’ having count 13555.

**Conclusions**

In this Round 1 of our project, we performed the tasks of word pre-processing, word tokenisation, Word Cloud generation, POS tagging and also deduced many inferences from them about the book while also learning in the process