CS21M015 (Data Scraping)

The task for this part of the project is to scrap data from the discussion from popular repositories and identify topics discussed. These discussions are basically a set trailing messages in open/closed pull requests/issues.

PyGithub library is used for performing this task. In the process first we created a personal access token in github account. The code for performing this task is in scerp.py file. Using the token, we connected to github repository Fenix which is from mozilla-mobile. The counter is set till 2000, after which the program will stop and safe the 2000 rows obtained in a dataframe to a file named fenix___.csv which is used in the upcoming phases for cleaning and analysis of the data.

While the data is being scraped there may be chance of the code being abruptly terminated for exceeding the number of requests a token can be used for. To prevent this loss of data, it is being saved in the except clauses.

CS21M011 (Data cleaning and preprocessing)

The issue_comments field has to be filtered very neatly for the analysis of comments. Most of the comments are having special symbols, Emojis, alpha-numeric characters, HTML tags and links also. So all these have to be removed to apply the preprocessing teechniques on the text. In addition to this, replace the shortcuts like isn't with is not, won't with will not etc.

While scraping the data, we got the issue_comments, issue_body fields in json format. So, this json format can be converted into plain text by applying the above cleaning techniques.

In the Preprocessing phase we did the following in the order below:-

1. Removing the json format
2. Removing the html tags
3. Remove any punctuations or limited set of special characters like , or . or # etc.
4. Check if the word is made up of only english letters and is not alpha-numeric
5. Check to see if the length of the word is greater than 2 (as it was researche d that there is no adjective in 2-letters)
6. Convert the word to lowercase
7. Remove Stopwords
8. Finally Snowball Stemming the word (better than Porter Stemming)

After cleaning is done, include the words which are not there in the list of stop words in English. These stop words are already included in nltk library.

Preprocessing:

Bag of words:

Applying this technique gave the no. of unique words as 1781 with the min_df=5. Means, we consider the word into Bag of words when only it appears for morethan 5 times in the corpus.

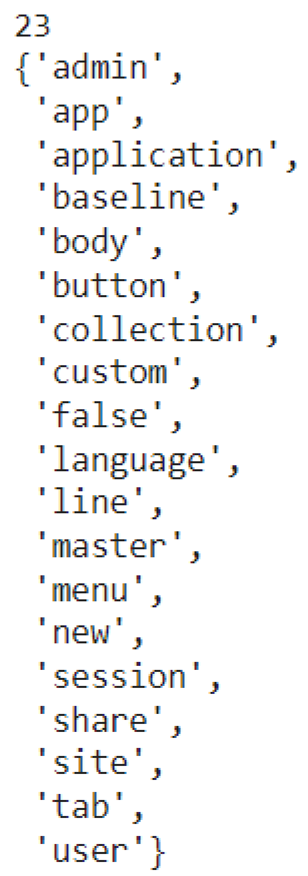
Bi-grams, Tri-grams, N-grams:

This technique is nothing but applying BoW on multiple words together. It takes combination of multiple words. Based on research, 5grams will give better analysis. Then we got 252 unique words in the corpus. Analysing this 252 words manually is tough. So, apply Lemmatization.

Lemmatization:

Lemmatization finds out the root word for a particular set of related words.
For example, consider the words as **tasty, taste, delicious, tasteful** etc. All these are mapped to a root word named **tast**.
Similarly, for King and Queen it can map them Male and Female respectively.

For this technique, we used pretrained model from spacy named ***en_core_web_md.*** This technique tokenizes the corpus and gives the list of root words.



```
23
{'admin',
 'app',
 'application',
 'baseline',
 'body',
 'button',
 'collection',
 'custom',
 'false',
 'language',
 'line',
 'master',
 'menu',
 'new',
 'session',
 'share',
 'site',
 'tab',
 'user'}
```

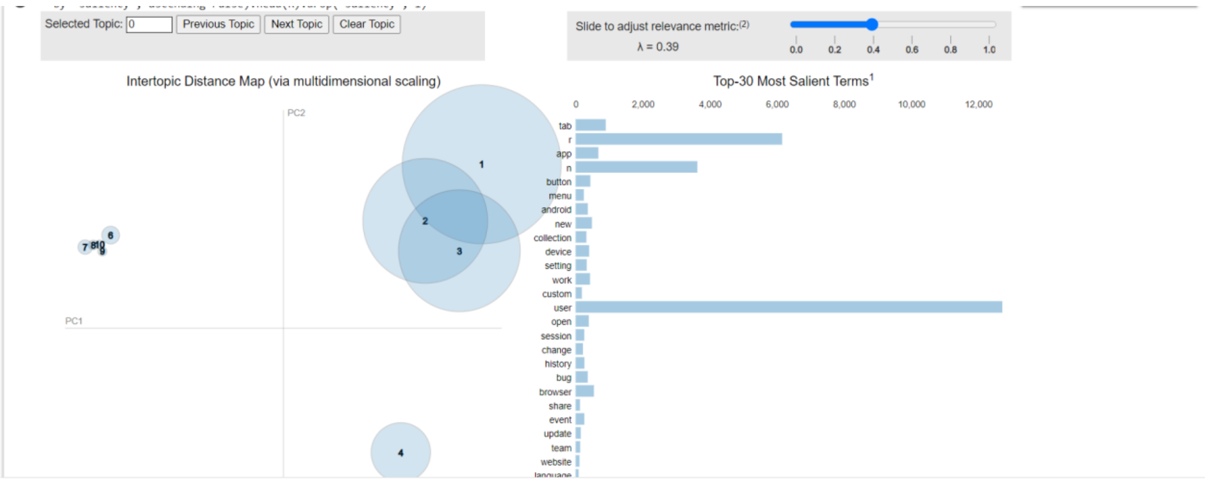
Visualization:

pyLDAvis is the library used for this. It stands for python Long Data Visualization. LDA model analyses the tokenized words from comments and gives the probability of each word in the sentence.

From this, we can fetch the words which have occurred more than the probaility of threshold. Threshold can be taken as the mean of the words in the corpus.

Genimvis:

This gives a pictorial representation (bar graph) for the most_discussed_words.



Selected Topic: 0

Slide to adjust relevance metric:(2)

λ = 0.39

Colab link:

https://colab.research.google.com/drive/1lPB3fXzTzol02dptTOFU223uLVnyKbNL#scrollTo=DoDV6AmJfhet