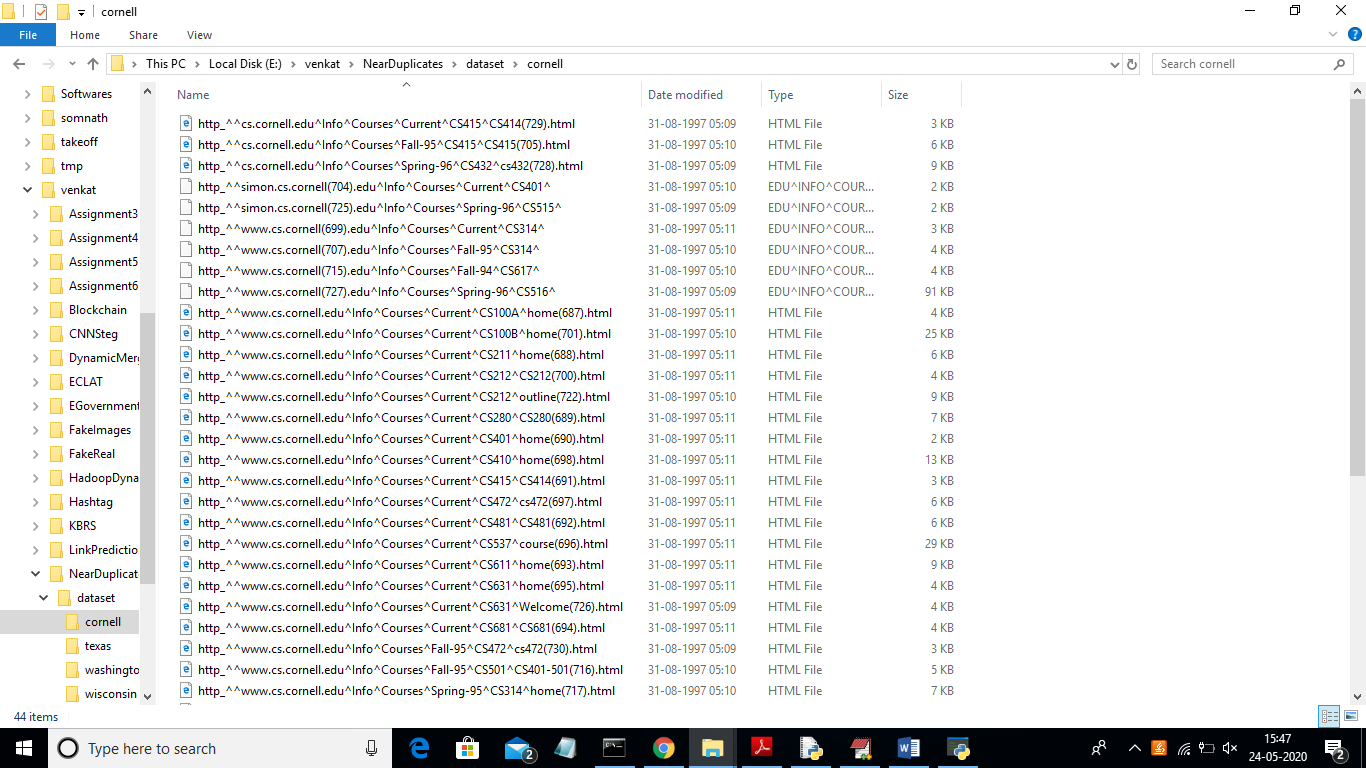
A Strategy for Near-Deduplication Web Documents Considering Both Domain &Size of the Document

In this project author is describing concept to de-duplicate web documents by using NLTK (Natural Language Tool Kit) and TF-IDF (Term Frequency-Inverse Document Frequency) technique. Now-a-days large amount of web documents are designing and there is a chance that some documents may be duplicate with one and other and if find and remove such duplicate documents then internet server storage can be minimize and while accessing web pages less no of web documents will be downloaded or render to user browser which can increase web pages accessing speed.

To implement this project we are using WEBKB dataset which consists of course web pages and by using this dataset we can find near duplicates documents and then it can be remove from server to decrease storage space. This dataset contains all HTML files. See below dataset screen shots



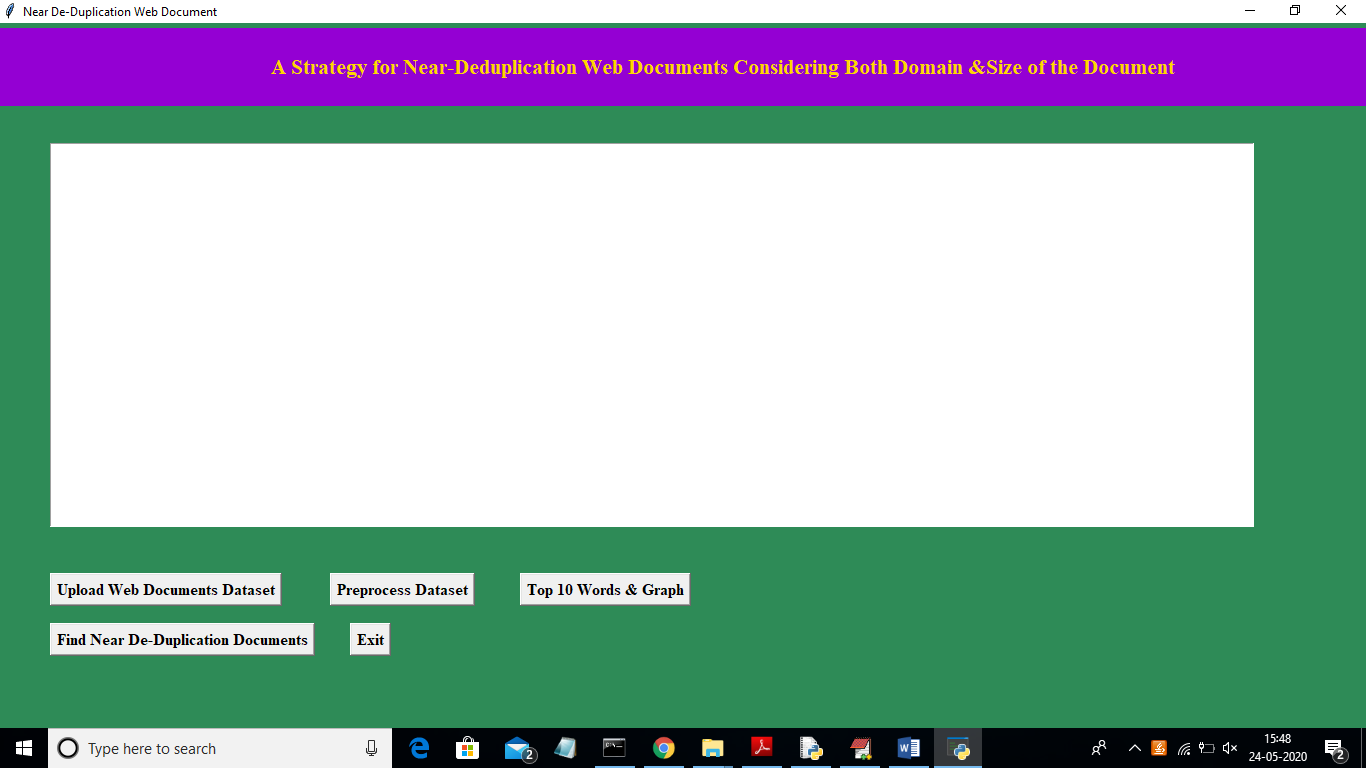
To implement this project we are using STEMMING, STOPWORDS removal, TF-IDF and TOKENIZER techniques.

This projects consists of following modules

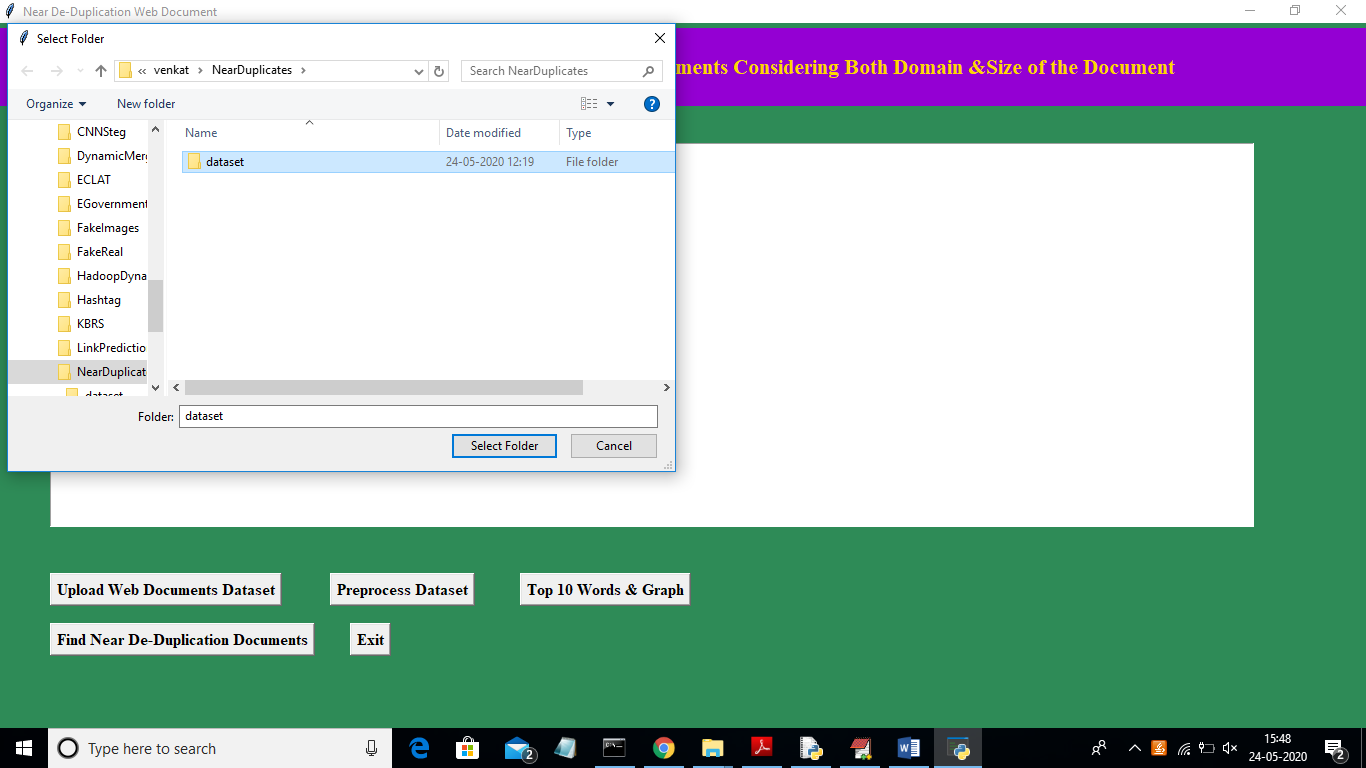
1. Upload Web Documents Dataset: using this module we will upload WEB Documents dataset to this application. Application read/parse html document using python JSOUP API and then remove stop words and special symbols from text. Apply stemming technique to get root words. Clean document will be appended to array.
2. Pre-process Dataset: Using this module we will convert text document array to TF-IDF values by using python TFIDF transformation class.
3. Top 10 Words & Graph: Using this module we will find top 10 high frequency occurrence words which means top 10 words which occur more times.
4. Find Near De-Duplication Documents: Using this module we will check similarity of one document with rest of the document and the two documents with high similarity will be consider as near duplicates. To find similarity between WEB documents we are using EUCLIDEAN DISTANCE formula.

SCREEN SHOTS

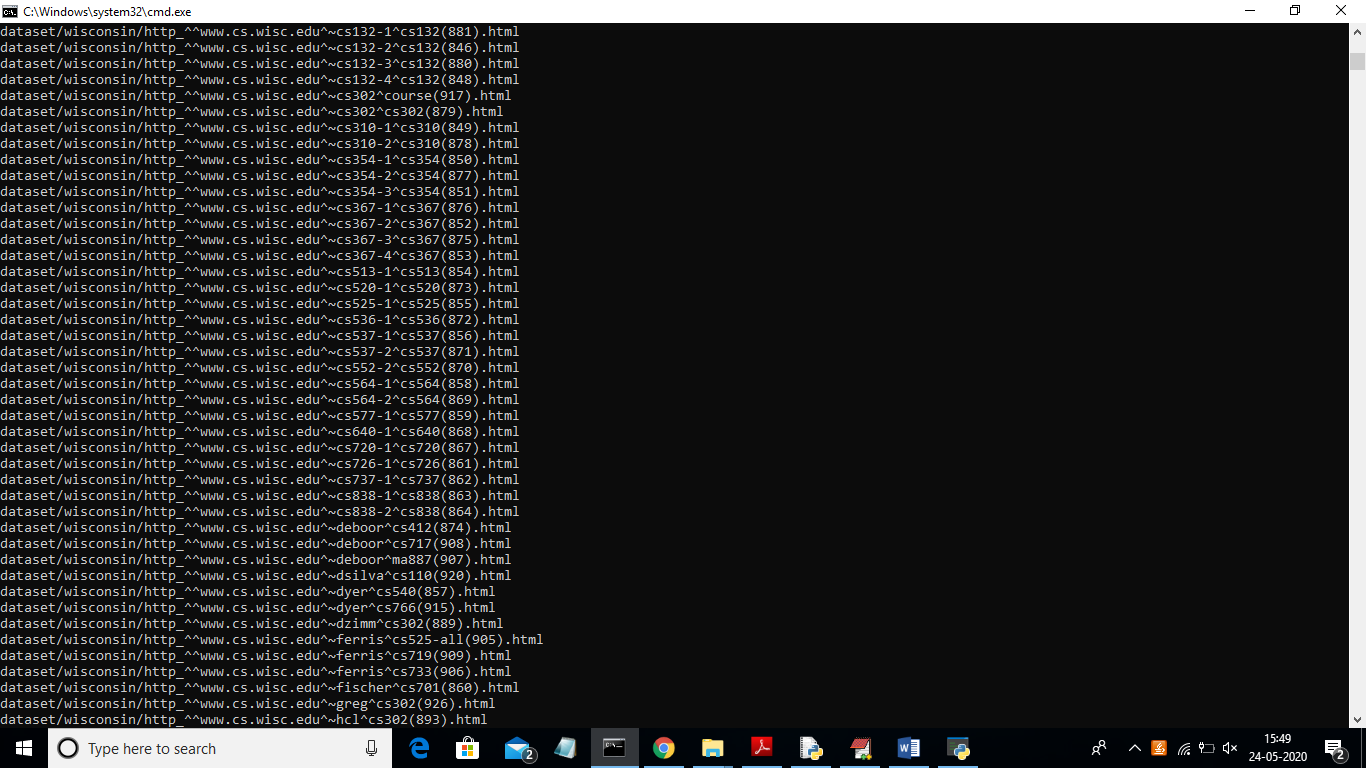
To run project double click on ‘run.bat’ file to get below screen



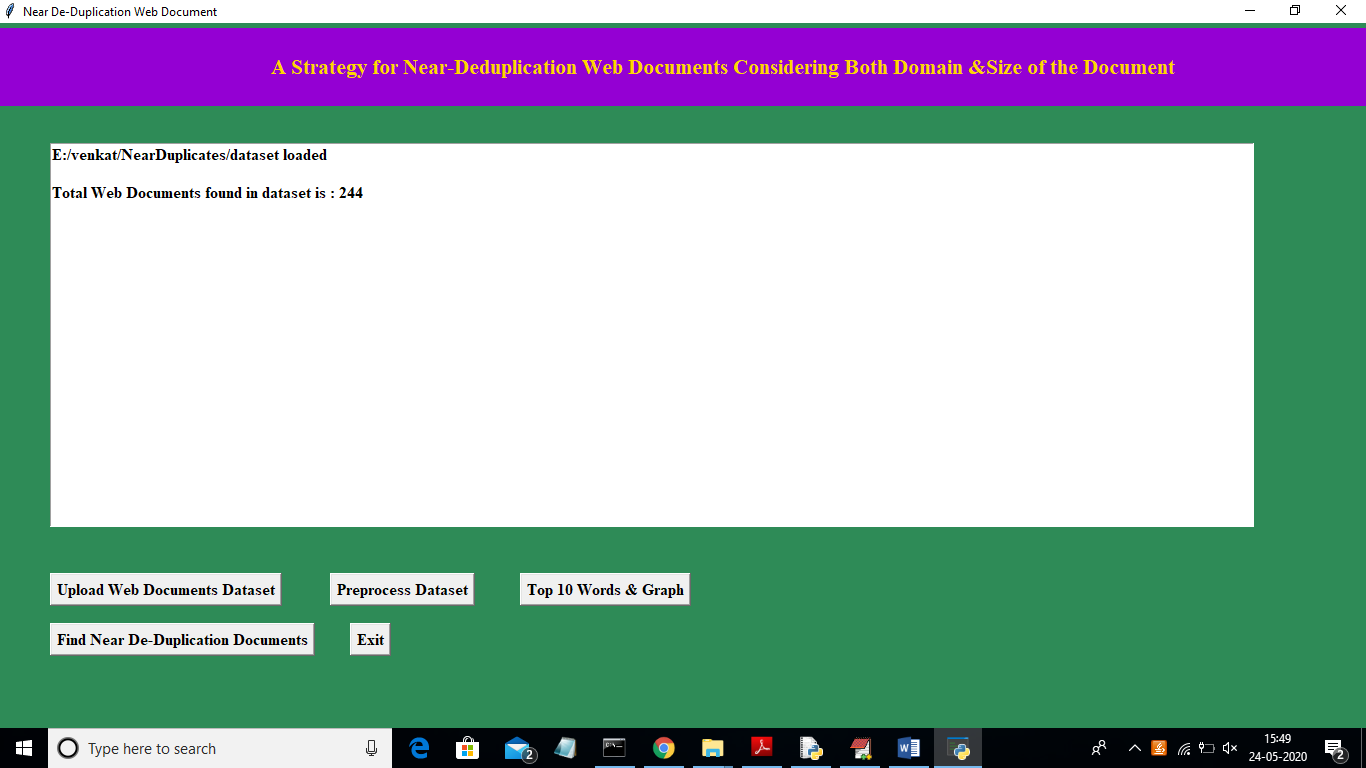
In above screen click on ‘Upload Web Document Dataset’ button and upload dataset folder



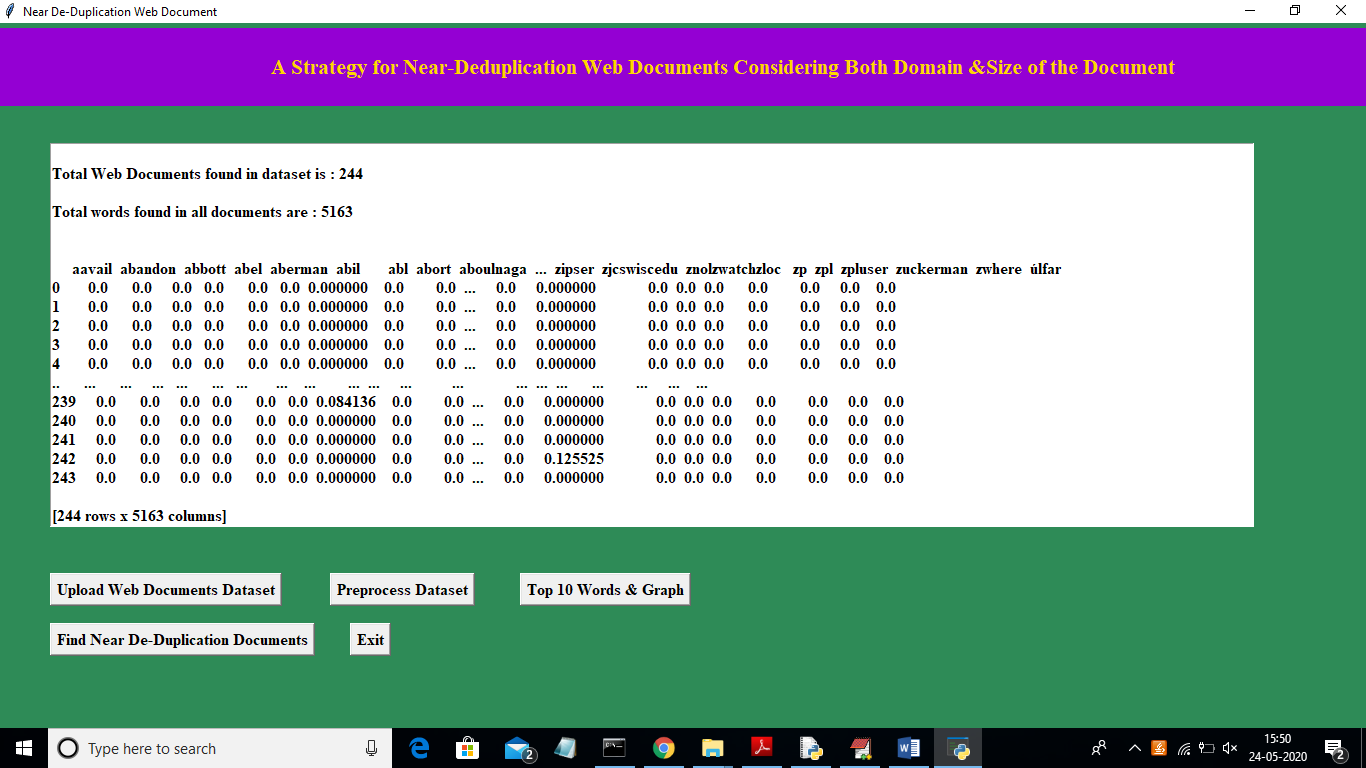
In above screen I am uploading ‘Dataset’ folder and now click on ‘Select Folder’ button to load dataset. All files reading we can see in console black screen



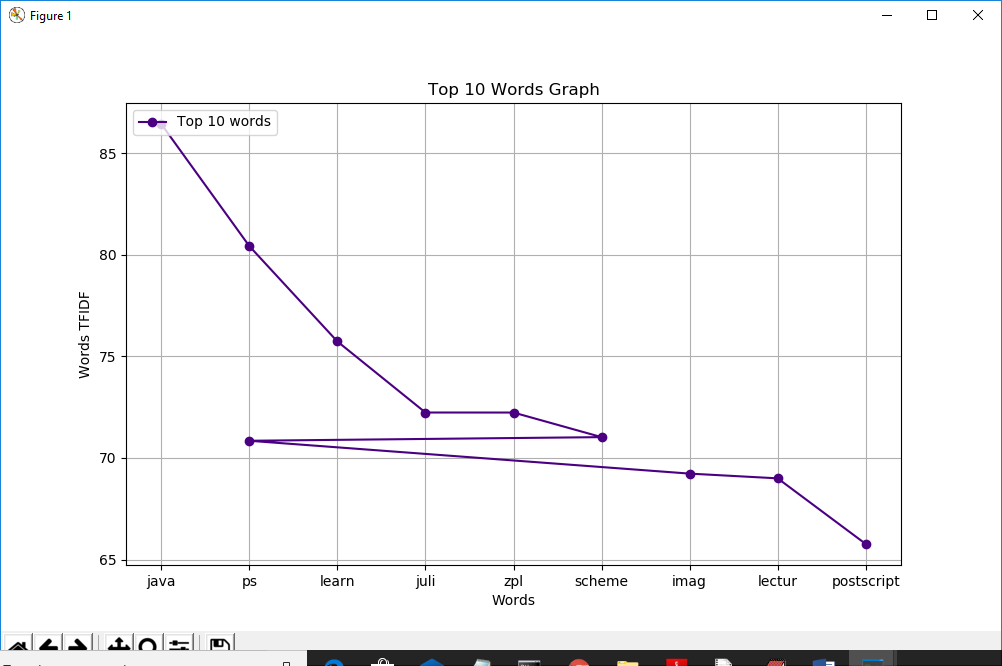
After loading dataset will get below screen



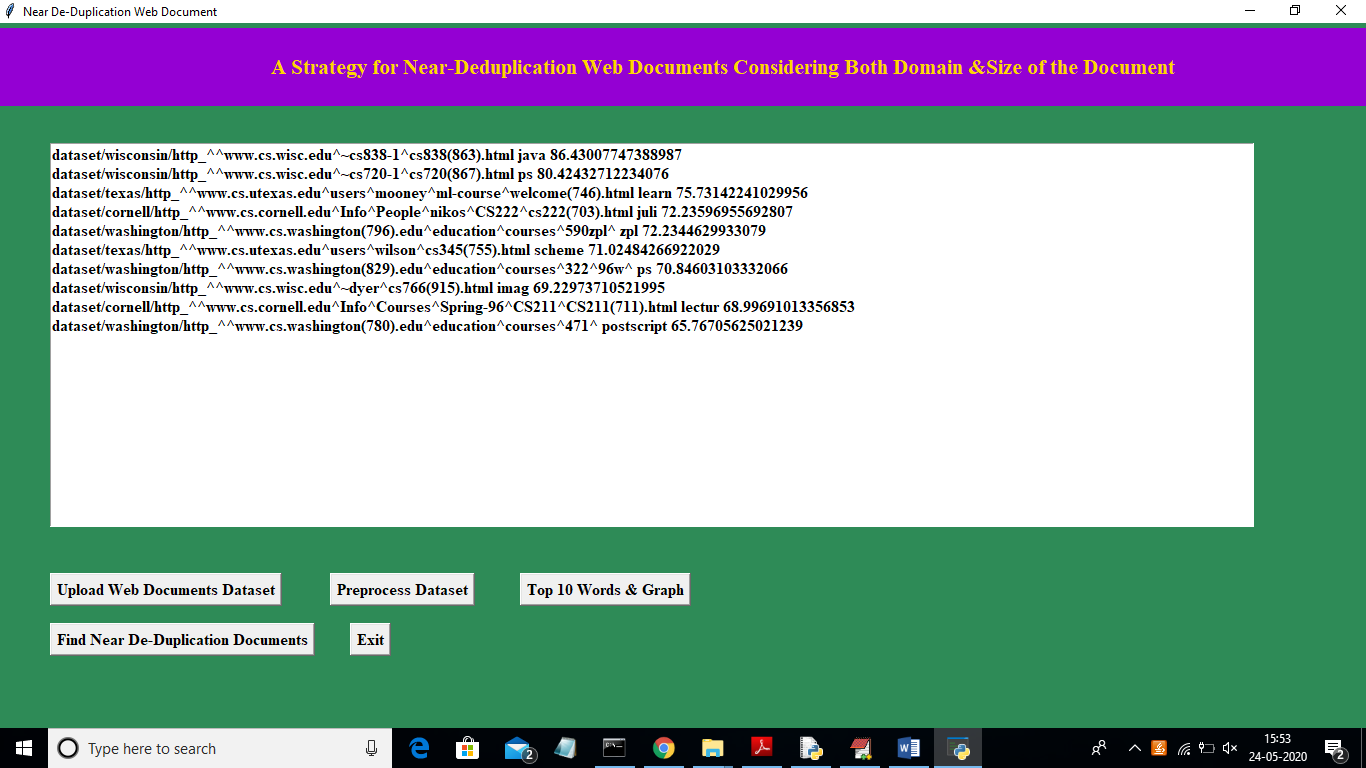
In above screen we can see dataset contains total 244 WEB documents. Now click on ‘Preprocess Dataset’ button to convert dataset into TF-IDF values



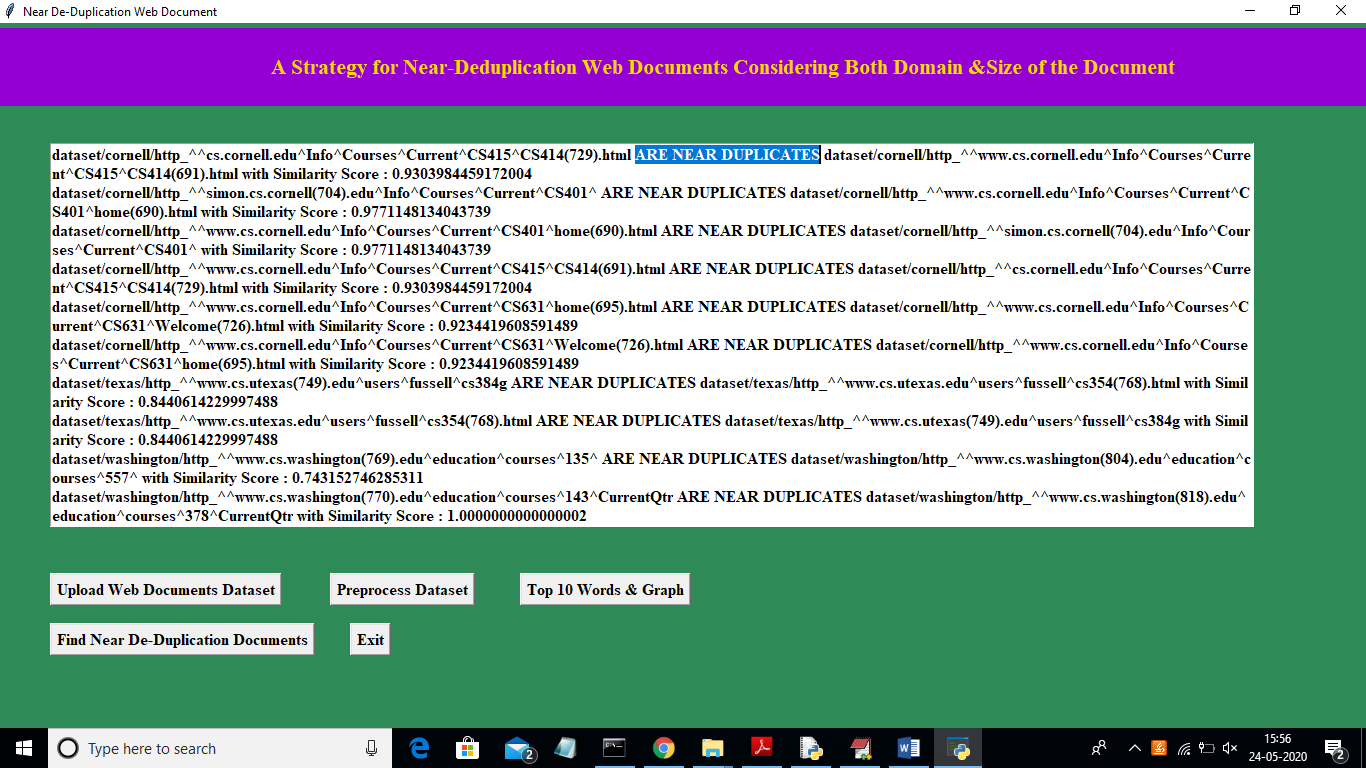
In above screen we can see dataset contains 244 documents and all documents contains 5163 unique words and below it we can see TF-IDF values for each word. It’s not possible to show all 5163 words TF-IDF values so I am showing few words in above screen. Now click on ‘Top 10 Words & Graph’ button to view top 10 frequent words.



In above graph x-axis represents words and y-axis represents frequency of thos words. Same we can see in below screen



In above screen each line has 3 words separated with space. First value contains file name in first line and second value after space is then frequent top 10 word and third value is the frequency of the word. In above screen in first line 86.43 is the frequency or TF-IDF value of word ‘java’. Similarly u can see for other words also. Now click on ‘Find Near De-Duplication Documents’ button to get all near duplicates files



In above screen between ‘ARE NEAR DUPLICATES’ we will find two documents names which are near duplicates and beside them we can see their similarity matching score.