19BCE2311 Gaurav Singh

Question:

Write a python program to

Find out the list of common and unique terms in the between the three text files attached (chess.txt, tennis.txt, cricket.txt) and print its count.

Apply stopword removal on those common and unique terms (using spacy), print its count and save the terms (after stopword removal) in index.txt file.

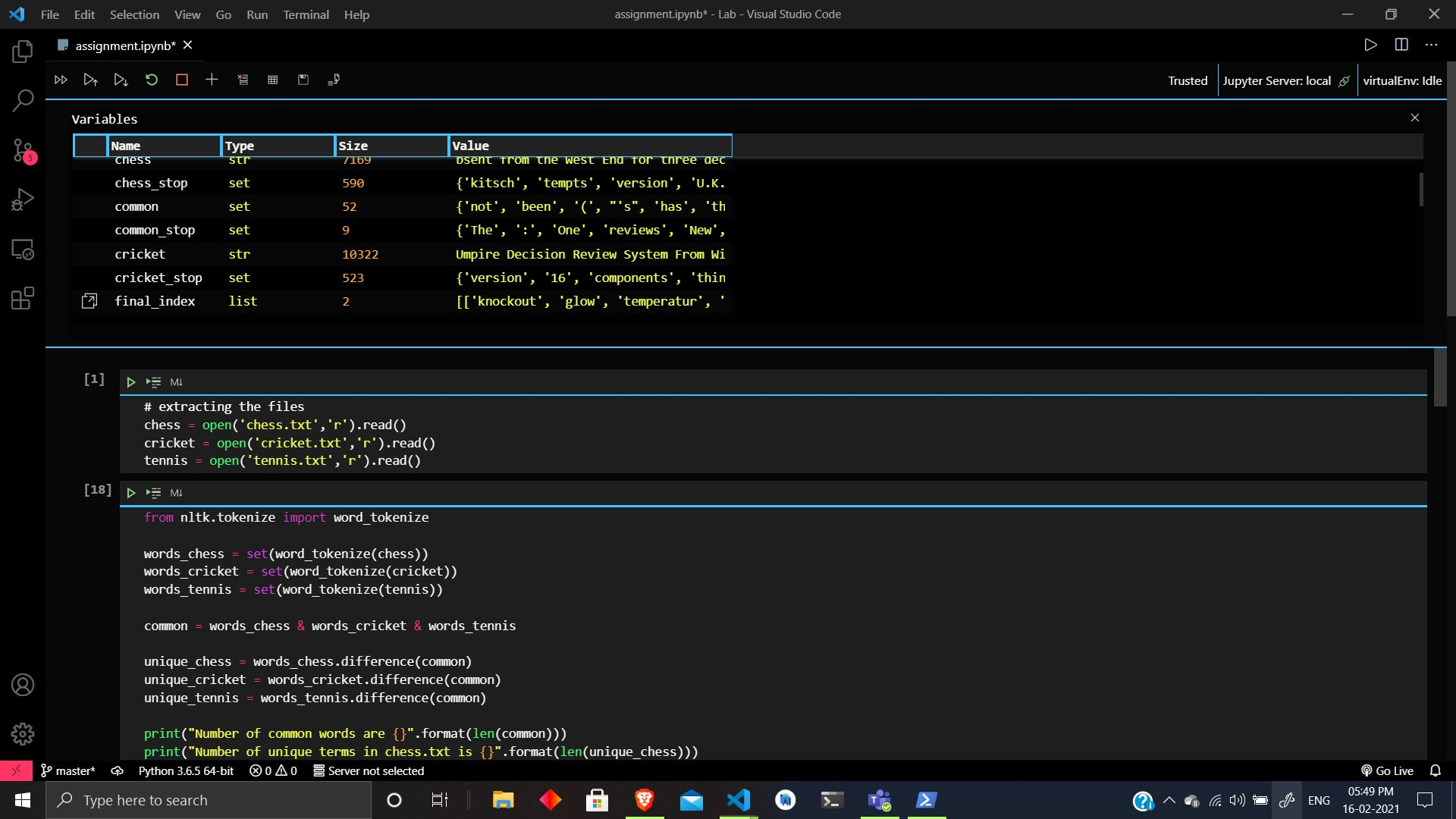
[List of additional Stop words to be considered = [dot, comma, single-quote, double quote, question mark, brackets [square, parentheses, curly, angle], exclamation mark]]

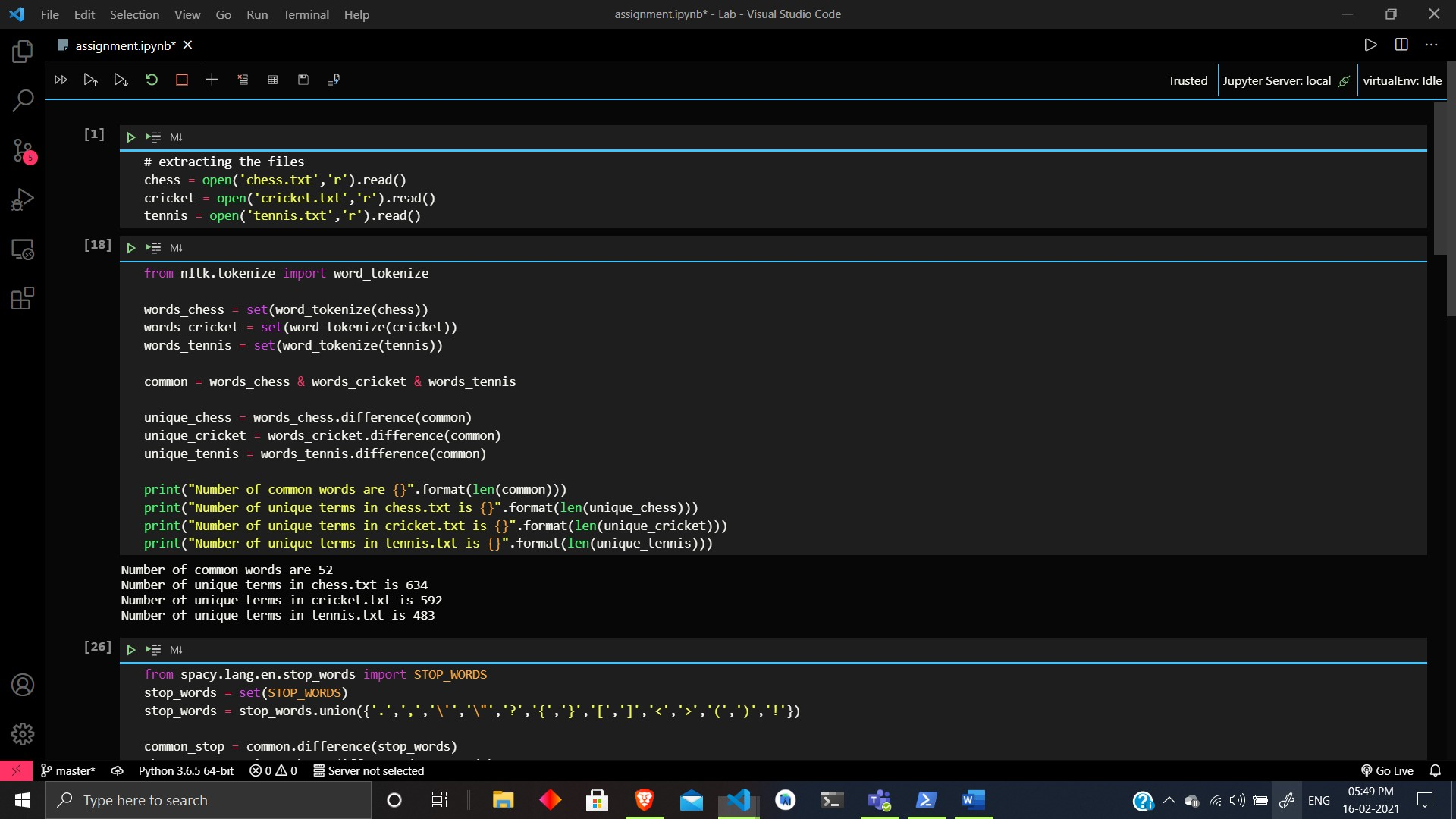
Apply Stemming and lemmatization on the terms present in index.txt file. Print the count of terms after applying stemming and lemmatization.

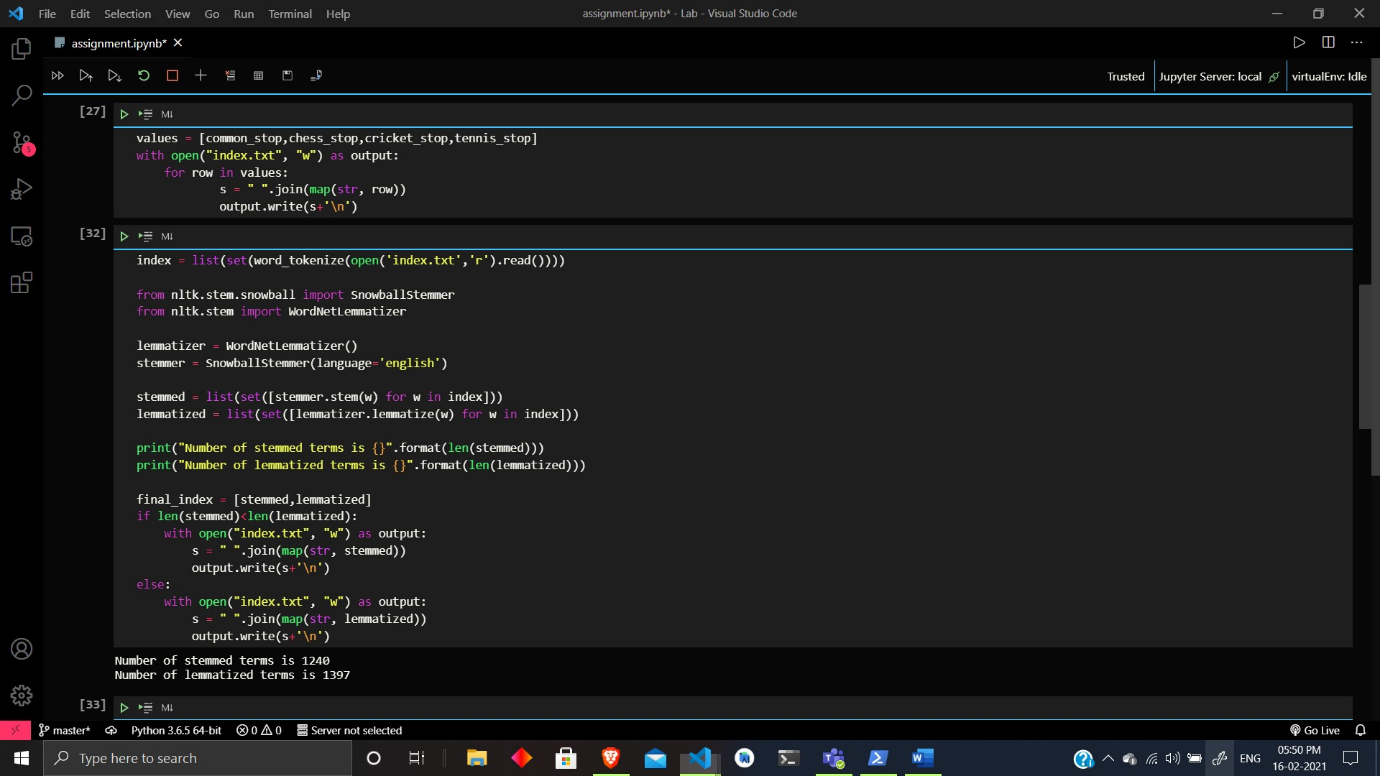
Replace the content of index.txt file by either stemmed terms or lemmatized term depending on its count.(lower count value should be considered for replacement) and rename the file name to final-index.txt

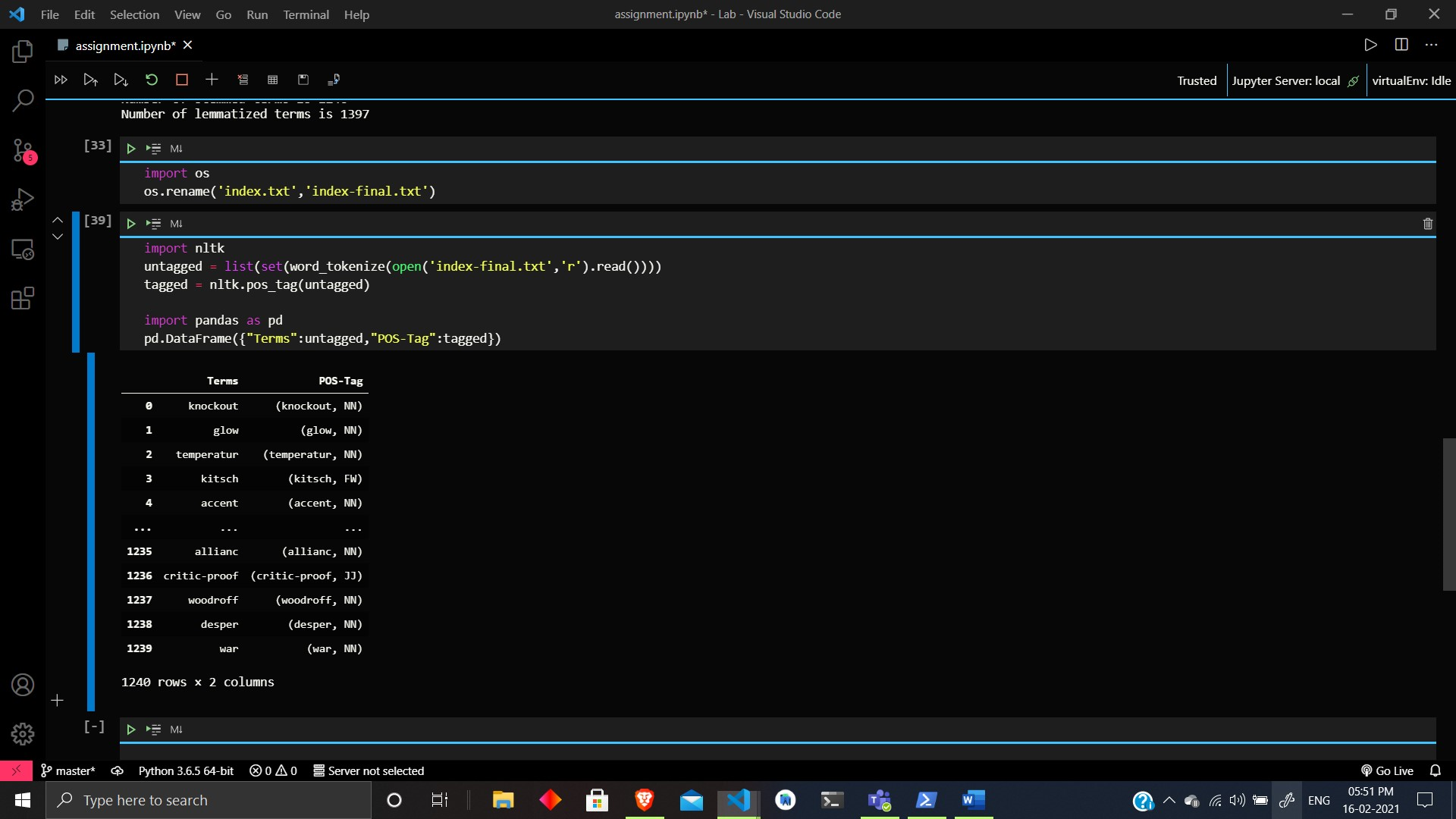
Print the POS tag of all the terms present in the final-index.txt file using pandas dataframe

Screenshots:









Code:

# extracting the files

chess = open('chess.txt','r').read()

cricket = open('cricket.txt','r').read()

tennis = open('tennis.txt','r').read()

from nltk.tokenize import word\_tokenize

words\_chess = set(word\_tokenize(chess))

words\_cricket = set(word\_tokenize(cricket))

words\_tennis = set(word\_tokenize(tennis))

common = words\_chess & words\_cricket & words\_tennis

unique\_chess = words\_chess.difference(common)

unique\_cricket = words\_cricket.difference(common)

unique\_tennis = words\_tennis.difference(common)

print("Number of common words are {}".format(len(common)))

print("Number of unique terms in chess.txt is {}".format(len(unique\_chess)))

print("Number of unique terms in cricket.txt is {}".format(len(unique\_cricket)))

print("Number of unique terms in tennis.txt is {}".format(len(unique\_tennis)))

from spacy.lang.en.stop\_words import STOP\_WORDS

stop\_words = set(STOP\_WORDS)

stop\_words = stop\_words.union({'.',',','\'','\"','?','{','}','[',']','<','>','(',')','!'})

common\_stop = common.difference(stop\_words)

chess\_stop = unique\_chess.difference(stop\_words)

cricket\_stop = unique\_cricket.difference(stop\_words)

tennis\_stop = unique\_tennis.difference(stop\_words)

print("Number of terms common after removal of stop words = {}".format(len(common\_stop)))

print("Number of terms unique in chess.txt after removal of stop words = {}".format(len(chess\_stop)))

print("Number of terms unique in cricket.txt after removal of stop words = {}".format(len(cricket\_stop)))

print("Number of terms unique in tennis.txt after removal of stop words = {}".format(len(tennis\_stop)))

values = [common\_stop,chess\_stop,cricket\_stop,tennis\_stop]

with open("index.txt", "w") as output:

    for row in values:

            s = " ".join(map(str, row))

            output.write(s+'\n')

index = list(set(word\_tokenize(open('index.txt','r').read())))

from nltk.stem.snowball import SnowballStemmer

from nltk.stem import WordNetLemmatizer

lemmatizer = WordNetLemmatizer()

stemmer = SnowballStemmer(language='english')

stemmed = list(set([stemmer.stem(w) for w in index]))

lemmatized = list(set([lemmatizer.lemmatize(w) for w in index]))

print("Number of stemmed terms is {}".format(len(stemmed)))

print("Number of lemmatized terms is {}".format(len(lemmatized)))

final\_index = [stemmed,lemmatized]

if len(stemmed)<len(lemmatized):

    with open("index.txt", "w") as output:

        s = " ".join(map(str, stemmed))

        output.write(s+'\n')

else:

    with open("index.txt", "w") as output:

        s = " ".join(map(str, lemmatized))

        output.write(s+'\n')

import os

os.rename('index.txt','index-final.txt')

import nltk

untagged = list(set(word\_tokenize(open('index-final.txt','r').read())))

tagged = nltk.pos\_tag(untagged)

import pandas as pd

pd.DataFrame({"Terms":untagged,"POS-Tag":tagged})