**Assignment-7 (Python)**

**1. WAP to scrape any web-site. Print the top 5 repeated words and count them and plot the graph of it.**

Sol.

import requests

from bs4 import BeautifulSoup as bs

import operator

from collections import Counter

import matplotlib.pyplot as plt

def start(url):

blacklist = ['[document]','noscript','header','html','meta','head', 'input','script','style']

wordlist=[]

source=requests.get(url).text

soup=bs(source,'html.parser')

text= soup.find\_all(text=True)

for t in text:

if t.parent.name not in blacklist:

word=t.lower().split()

wordlist.append(format(word))

clean\_wordlist(wordlist)

def clean\_wordlist(wordlist):

clean\_list =[]

for word in wordlist:

symbols = '!@#$%^&\*()\_-+={[}]|\;:"<>?/.,\ '

for i in range (0, len(symbols)):

word = word.replace(symbols[i], '')

if len(word) > 0:

clean\_list.append(word)

for x in clean\_list:

if x == '\'\'':

clean\_list.remove(x)

create\_dictionary(clean\_list)

def create\_dictionary(clean\_list):

word\_count = {}

for word in clean\_list:

if word in word\_count:

word\_count[word] += 1

else:

word\_count[word] = 1

c = Counter(word\_count)

top = c.most\_common(5)

plotGraph(top)

def plotGraph(top):

keyword=[]

count=[]

for t in top:

keyword.append(t[0])

count.append(t[1])

# ax = fig.add\_axes([0,0,1,1])

plt.bar(keyword,count,align='center',alpha=0.5)

plt.title('Top 5 words on the website and their count')

plt.xlabel("Words")

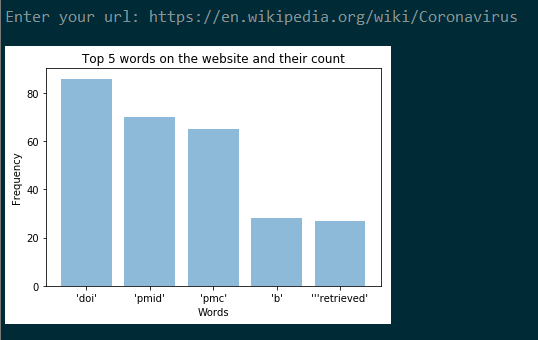
plt.ylabel("Frequency")

plt.show()

if \_\_name\_\_ == '\_\_main\_\_':

url=str(input("Enter your url: "))

start(url)



**2. WAP to do a sentiment analysis of any word entered by the user in voice command.**

Sol.

import speech\_recognition as sr

from textblob import TextBlob

sample\_rate = 48000

chunk\_size = 2048

r = sr.Recognizer()

with sr.Microphone(device\_index = 1, sample\_rate = sample\_rate,

chunk\_size = chunk\_size) as source:

r.adjust\_for\_ambient\_noise(source)

print("Say Something")

audio = r.listen(source)

try:

text = r.recognize\_google(audio)

print("you said: " + text)

blob=TextBlob(text)

print(format(blob.sentiment))

print("In Polarity, 0 indicates neutral, +1 indicates a very positive sentiment and -1 represents a very negative sentiment. ")

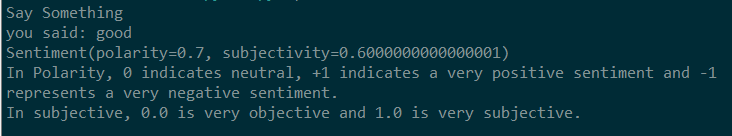
print("In subjective, 0.0 is very objective and 1.0 is very subjective.")

except sr.UnknownValueError:

print("Google Speech Recognition could not understand audio")

except sr.RequestError as e:

print("Could not request results from Google Speech Recognition service; {0}".format(e))



**3. WAP to pin different locations of Fruit Farms on the map and plot a pie chart according to states.**

Sol.

import csv

import requests

from bs4 import BeautifulSoup

import pandas as pd

import matplotlib.pyplot as plt

from geopy.geocoders import Nominatim

import folium

url = 'https://www.mapsofindia.com/indiaagriculture/fruits-map/'

response = requests.get(url)

html = response.content

heading=[]

soup = BeautifulSoup(html,'html.parser')

table = soup.find('table', attrs={'class': 'tableizer-table'})

tableHeader=table.find\_all('th')

for x in tableHeader:

heading.append(x.text)

list\_of\_rows = []

for row in table.findAll('tr')[1:]:

list\_of\_cells = []

for cell in row.findAll('td'):

text = cell.text.replace('&nbsp;', '')

list\_of\_cells.append(text)

list\_of\_rows.append(list\_of\_cells)

outfile = open("./fruit.csv", "w")

writer = csv.writer(outfile)

writer.writerow(heading)

writer.writerows(list\_of\_rows)

outfile.close()

df = pd.read\_csv ('fruit.csv')

state\_name = df[heading[1]]

production = df[heading[2]]

colors = ["#1f77b4", "#8c564b", "#ff7f0e", "#2ca02c", "#d62728"]

plt.title("Production of fruits in different states")

plt.pie(production , labels=state\_name , colors=colors,

autopct='%1.1f%%', shadow=True, startangle=140)

plt.show()

geolocator = Nominatim(user\_agent='myGeocoder', timeout=10)

country='India'

dict={}

for x in state\_name:

print("\n",x,"\n")

loc = geolocator.geocode(x+','+ country)

dict[x]=loc.latitude,loc.longitude

m=folium.Map(location=list(dict['Gujarat']),zoom\_start=5,tiles='Stamen Terrain')

for x in dict:

folium.Marker(

location=list(dict[x]),

popup=x,

icon=folium.Icon(color='red')

).add\_to(m)

m.save('index.html')

