YAAW

YIELDING ACCOMPLISHED AFRICAN WOMEN

TOPIC: A Python Project to Capture 100-1000 Tweets from a given Country and Store the Tweets in a File

TEAM NAME: ASAS

FACILITATORS: Richmond & Brymi

TEAM MEMBERS:

Francisca Ama Asiedu

Chima Stella Ukachi

Araba Affran - Annan

Sesinam Gagblezu-Alomatu

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION:

In today's world, There are billions of data available on the internet and manually accessing them can take days and even months to access. It is due to this that the automatic collection or extraction of data for various purposes have been on the increase.

BRIEF OVERVIEW OF OUR PROJECT:

This project focuses on capturing tweets from "Twitter" and storing in a file. These tweets are gotten from a given country. The minimum number of tweets that can be captured is 100 and the maximum number of tweets that can be captured is 1000. To extract these data, Python libraries such as os, csv, tweepy and pycountry are used. The list data structure was also used to implement this project to handle the duplicates.

DEFINITIONS:

The definitions of some of the concepts to be used are as follows:

Web Scraping: This is the collection and extraction of data from the web. This data can be used for various purposes based on their need. This data can also be scrapped manually or automatically(using software tools). The best way to extract data is via software tools such as Python libraries.

APIs and Objects: Application Programming Interface is a tool set that a programmer can use in creating a software and an object is a function, variable or data structure that can be referenced in memory by an identifier.

File Storage: This can be defined as a hierarchical storage methodology used to organize and store data on a computing device.

CHAPTER TWO

This chapter contains the tools/frameworks, code and implementation as follows:

ARCHITECTURE:

1. **Tools/Frameworks/Libraries Used in Development:**

Tweepy: This is a python library for accessing Twitter API. It granted us access to the details for using OAuth required by the Twitter API. It also allowed us to capture tweets from different countries.

Pycountry: This library gave us access to a database consisting of existing countries.

Os: We used the os module to create the directory to store the files that have tweets in them.

Csv: The csv(Comma Separated Values) module allowed us to store tweets in csv file by writing to the file in rows.

2. **Implementation:**

❖ **Tweet_capture.py:**

- **def menu():**

```
def menu():
  print('a: Capture tweets from any given country')
  print('b: Read existing tweets from file')
  print('Choose an option: ')
  choice = str(input(""))
  if choice == "a":
      capture_tweets()

  elif choice == "b":
      read_tweets()

  else:
```

```
    print('select a valid option')
    return menu()
```

The code starts by displaying a list of options for the user to select from. If the choice ==”a”, it goes ahead to call the capture_tweets() function. If the user's choice == “b”, it goes ahead to call the read_tweets() function. If the user inputs a choice other than “a” and “b”, it informs the user to select a valid option. (These functions will be explained below).

- **def capture_tweets():**

```
def capture_tweets():
    results = name_country()  # a list containing the country's name and tweet count
    tweet_timestamps = []
    tweet_texts = []

    print(f'Capturing tweets from {results[0]}...')

    # get the tweets and their corresponding timestamps into lists, removing duplicates
    for tweet in tweepy.Cursor(api.search, q=results[0]).items(results[2]):
        t_timestamp = tweet.created_at
        t_text = tweet.text.encode('utf-8')

        if not tweet_texts.__contains__(t_text):  # checks if the tweet has already been captured
            tweet_timestamps.append(t_timestamp)
            tweet_texts.append(t_text)

        # if the captured tweets are equal to the count specified continue
        if len(tweet_texts) == results[1]:
            break

    __dir = './Tweets'
    if os.path.exists(__dir):
        pass
    else:
        os.mkdir(__dir)  # creates Tweets directory

    csv_file = open(f'{__dir}/{results[0]}.csv', 'a')
    csv_writer = csv.writer(csv_file)

    for i in range(len(tweet_texts)):  # writes captured tweets to the file
        csv_writer.writerow([tweet_timestamps[i], tweet_texts[i]])

    print('Done!')
```

As the name implies, this is the parent function that captures the tweets. The function starts by calling the name_country() function which asks the user to input the country they would like to capture tweets from. The timestamp displays the time the tweet was created. After the country name has been verified, it begins the process of extracting the tweets. To help make pagination easier and require less code, tweepy uses the cursor object. This inbuilt tweepy function helps in narrowing our search.

The function also studies the case of duplicates whereby it checks if the tweets have already been captured. It also checks if the captured tweets are equal to the specified count.

A directory called ‘./Tweets’ is then created. This directory is the location where all the captured tweets separated by countries are stored. The files in the ‘./Tweets’ directory are stored using this format: '{__dir}/{results[0]}.csv'. For example, ./Tweets/Ghana.csv .

- **def read_tweets():**

```
def read_tweets():
  filename = input('Kindly enter the filename: ')
  if os.path.isfile(f'./Tweets/{filename.title()}.csv'):
      with open(f'./Tweets/{filename.title()}.csv', 'r') as file:
          file.read()
          print(file.read())
  else:
      print('File does not exist!')
      read_tweets()
```

This function is to read the captured tweets in the file. It starts by requesting for the specific file name. If it doesn't exist, a recursion occurs.i.e the read_tweets() function is called again. If the file name exists, it reads from the file and displays the result to the user.

- **def name_country():**

```
def name_country():
    countries = [i.name for i in list(pycountry.countries)]
    country_name = str(input("Please specify country name: ")).title().strip()
    if country_name in countries:
        return tweet_count(country_name)
    else:
        print("Country doesn't exist!")
        return name_country()
```

This function receives the country_name from the user. It then verifies using the pycountry library to ensure that the country input corresponds to an existing country else it displays an error message and using recursion, calls the name_country () function.

```
def tweet_count(country_name):
    try:
        count = int(input("Please specify number of tweets: "))
        excess_count = count * int(1.50 * count)  # increases the count by 50%
    except ValueError:
        print("Please enter a number!")
        return tweet_count(country_name)
    else:
        if (count >= 100) and (count <= 1000):
            return [country_name, count, excess_count]
        else:
            print("The number of tweets must be between 100 and 1000!.")
            return tweet_count(country_name)
```

This function receives the tweet_count from the user. The excess_count variable increases the count by 50%. The function catches the ValueError exception to ensure that only integer values are passed. The count must be between 100 and 1000 else displays an error message and recalls the function tweet_count().

- ❖ **main.py:**

```
from capture_tweet import tweet_capture

tweet_capture.menu()
```

The main.py simply imports the tweet_capture.py file from the capture_tweet package and calling the menu() function from the imported file runs the entire program.

3. Code Result:

1.

a: Capture tweets from any given country
b: Read existing tweets from file
Choose an option:
a
Please specify country name: Nigeria
Please specify number of tweets: 100
Capturing tweets from Nigeria...
Done!

Process finished with exit code 0

2020-08-27 16:11:04,b'RT @Nawas_masud: I PLEDGE TO NIGERIA MY COUNTRY. \xf0\x9f\xa4\x94\xf0\x9f\x99\x8f https://t.co/yqBJKM4TUu'

2020-08-27 16:11:04,"b'RT @CocoTennie: ""Social media is one of the fastest way to pass out information, so we need to continue advocating and creating awareness o\xe2\x80\xa6'"

2020-08-27 16:11:04,"b'RT @BashirAhmaad: The NCDC latest report on COVID-19 in Nigeria\n\n391,502 samples tested\n\n53,021 confirmed cases\n\n40,281 discharged cases\n\n1\xe2\x80\xa6'"

2020-08-27 16:11:04,b'RT @cuppymusic: Told @AppleMusic how SCARY putting music out in Nigeria can be! \xf0\x9f\x8e\xb6\xf0\x9f\xa7\x81 PROUD OF MYSELF! \xf0\x9f\x87\xb3\xf0\x9f\x87\xac #OriginalCopy https://t.co/4npKfKUP2p'

2020-08-27 16:11:03,b'RT @beverlyadaeze: Dogs in Nigeria are dealing with trauma too \xf0\x9f\x98\x82\xf0\x9f\x98\x82 https://t.co/VMXFm2hk2u'