Author: Nick Caravias
Email: ngc005@bucknell.edu
Instructor: Dr. Christian Howard-Sukhil

*Project TwitLit*: Fall 2020
Digital Humanities @ Bucknell University

Welcome to *Project TwitLit*'s CSV Scraper! This script was developed for the Digital Humanities project, *Project TwitLit*, in order to provide textual analysis for tweets pulled from Twitter's API. This script, however, can be used to analyze any csv file containing language text.

### Functions (If you are looking to use this script for TwitLit research skip this section)

This python script is an easy-to-use tool that enables computational textual analysis on scraped Twitter data. I had a few main goals in writing this script.

First, I wanted this script to be easy to use for anyone tasked with analyzing large csv files. To do so, this README file has in-depth instructions on how to use the script, including installations. Additionally, there are built-in instructions in the script that will print in the command terminal to ensure you are taking the right steps.

Second, I wanted to be able save pertinent textual analyses in one file. This script pulls four pieces of information:

1. Word frequency
2. Collation, words that occur together
3. Sentiment analysis
4. Language detection and frequency

The word frequency and collation features of this create a dictionary of the words in this step. I opened the csv file as a list of tuples to work with in this script.

## How to Use the Script

### Installations

There are a few downloads you will need to make before using this script.

First, make sure you have Python downloaded on your computer. You can check to see if this is on your computer by going to your command line terminal and typing

```
$ python —version
```

You should see an output like this:

```
nickcaravias$ python --version
Python 3.7.4
```

If this is not your output, then you should check out this link to install python on your computer: https://realpython.com/installing-python/

Next, we will need to make sure that you have language detector package installed. This is not a standard utility module in Python, so you will need to install this externally. Go to your command line and enter:

```
$ pip install langdetect
```

**Running in Command Line**

After installation, you are ready to use your script! Below is a step-by-step process. NB: If you are running this in a Python IDE, skip to step 4.

1. Enter your command line terminal.

2. Make sure that all of your files are in the same folder. This is important because later in the script's usage if you enter a file name that is not in the same file as a script you will not be able to find the csv file. There are two ways to do this. You can use finder, create a folder, and move this downloaded script along with the csv files you want to scrape into that folder. Or use command line to move these files. To do this enter the commands:

```
$ mkdir Twit_Lit
```

(this makes a folder and you can choose whatever name you want

```
$ mv [full name of your csv file] Twit_Lit
$ mv Twit_csv_Scraper.py Twit_Lit
```

(Move the Python script and csv files to your document)

```
$ cd Twit_Lit
$ ls
```

(Check to see that they are both in the folder, the names of the csv files and the Python script should be printed)

From here, let's start pulling information from the csv files using the python script!

3. Make sure your current directory is the previous file we created or the one in which you had previously stored the files. Run the script with the command:

```
$ python Twit_csv_Scraper.py
```

4. Next you will be prompted with the question:

```
Enter the name of the csv file:
```

Just make sure to include the .csv part of the file name. After this input is returned, the script will start to open the csv file and organize it into a readable form for the textual analysis. This can take a long time if it is a long csv file.

5. The next prompted questions are:

```
How many common words do you want to see?
```

and then…

```
Do you want to filter out common words? (answer only yes/no)
```

Only input your number in digits and a yes or no answer in lower case. The code is looking for these options.

6. Next you will be asked how many frequently linked words you want to see with the prompt:

```
How many frequently linked words do you want to see?
```

After this you are all done! A file should be created with all of the results.