**Assignment 3 (Group work): Building a Crowdsourced Recommendation System**

**Date handed out: October 18, 2018, Date due: November 2, 2018 by 11:59 p.m.**

**High level description:** The objective of this group assignment is to create the building blocks of a crowdsourced recommendation system. This recommendation system should accept user inputs about desired attributes of a restaurant and come up with 3 recommendations.

**Task A.** Use Web Scraper (free extension for Chrome, <http://webscraper.io/>) to extract 3-4k reviews of restaurants from Yelp (see primer posted on Canvas in Assignment 3 folder on how to use web scraper). On Yelp.com, specify the location – e.g., Austin downtown, and type of food (e.g., Chinese or Mexican food). Scrape the text reviews and well as the ratings (the number of stars) provided by the users (no need to scrape pictures of food and other details). The CSV output file should be processed to retain 3 columns – the name of the restaurant (Restaurant\_name), reviews (Restaurant\_review), and Rating.

**Task B.** Assume that a customer who will be using this recommendation system has specified 4 desirable attributes in a restaurant s/he is looking for – (i) service (e.g., speed, friendliness, etc.), (ii) food (e.g., quality, taste, etc.), (iii) price and (iv) location (e.g., parking, easy to find, drive through, etc.).

**Task C.** Perform a **similarity** analysis with the attribute set (service, food, price and location) and the reviews. See guidelines below on how to do this. From the output file, choose 200 reviews that have the highest similarity scores with the attribute set above.

**Task D.** Perform sentiment analysis on the 200 reviews and sort them (high to low) by the sentiment scores.

**Task E.** Based on tasks C and D, **recommend 3 restaurants** to the customer. Note that in task D, multiple reviews may refer to the same restaurant.

**Task F.** How would your recommendations differ if you ignored the similarity and sentiment scores and simply recommended the 3 highest rated restaurants from your entire dataset? To do this, you need to calculate the average rating (# stars like 3.5 or 4) for each restaurant mentioned (remember each restaurant in your data has many reviews, and you need to get the average rating for each restaurant). Would these three restaurants meet the requirements of the user looking for recommendations? Why or why not? Justify your answer. Use the ratings data, similarity and sentiment values to answer this question.

Your submission should include a word or pdf file showing

1. Names of all team members **inside** the document
2. Which food you searched for and the total number of reviews scraped
3. The average sentiment and similarity scores for the three restaurants you recommended in task E.
4. Your analyses and answer to task F. Make sure you show the ratings, similarity scores and sentiments for the restaurants you recommend in tasks E and F.

**Guidelines**

1. Follow the primer for web scraper on Canvas (in the Assignment folder) to scrape the Yelp data. Do not choose an esoteric or exotic food category where there may be just a few restaurants. Austin is a big city today, but it is not, say, New York city in terms of the types of food available.
2. Once you get the data in the csv format, retain three columns, Restaurant\_name, Restaurant\_review and Rating. Save the file as an Excel file, e.g., Yelp\_reviews.xlsx since the python scripts wants the data file to be in .xlsx format.
3. Use the attributes.txt file to specify the requirements of the user (e.g., service, price, etc.) – **one attribute per line**, no commas.
4. You will need to run similarity\_spacy.py for the similarity analysis. This script compares the words in the attributes.txt file with the reviews, and creates a similarity score between 0 and 1 for each review. Before you can run similarity\_spacy.py, you need to do the following on PythonAnywhere.com:

**Install spacy by using the following command in bash:**

pip3.6 install --user https://github.com/explosion/spacy-models/releases/download/en\_core\_web\_sm-1.2.0/en\_core\_web\_sm-1.2.0.tar.gz#egg=en\_core\_web\_sm-1.2.0

**Link the library by using the following command in bash:**

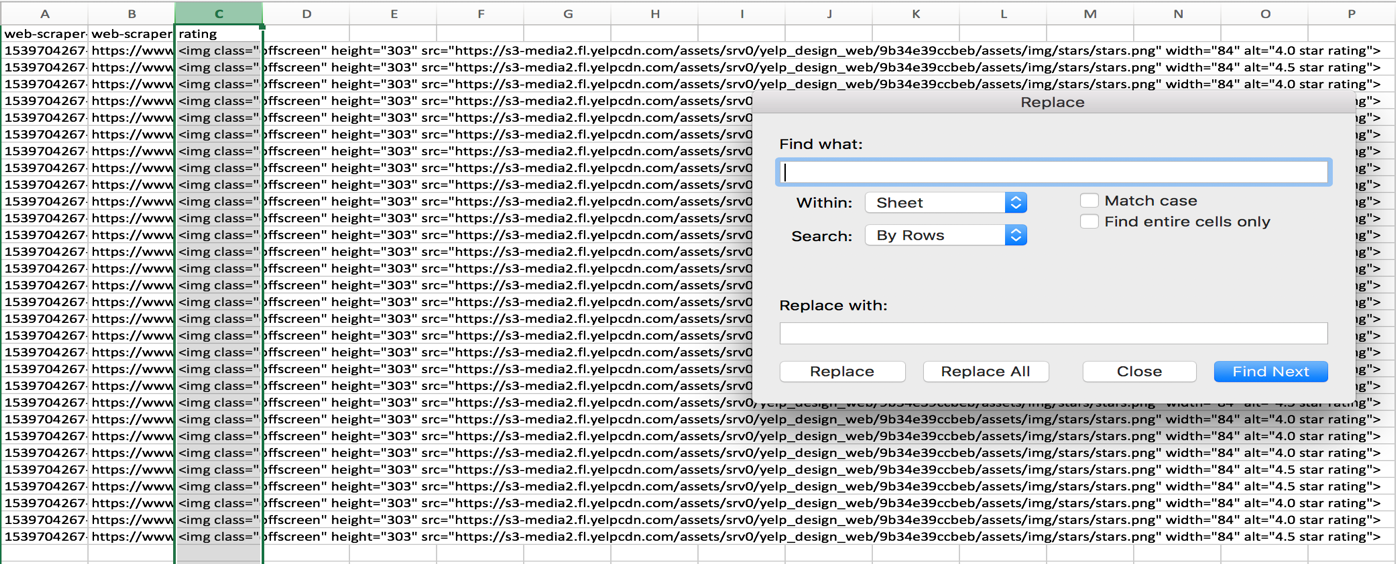
python3.6 -m spacy download en

**The input should be changed on line 32 as needed. Currently the input data file name is shown as yelp\_reviews.xlsx:**

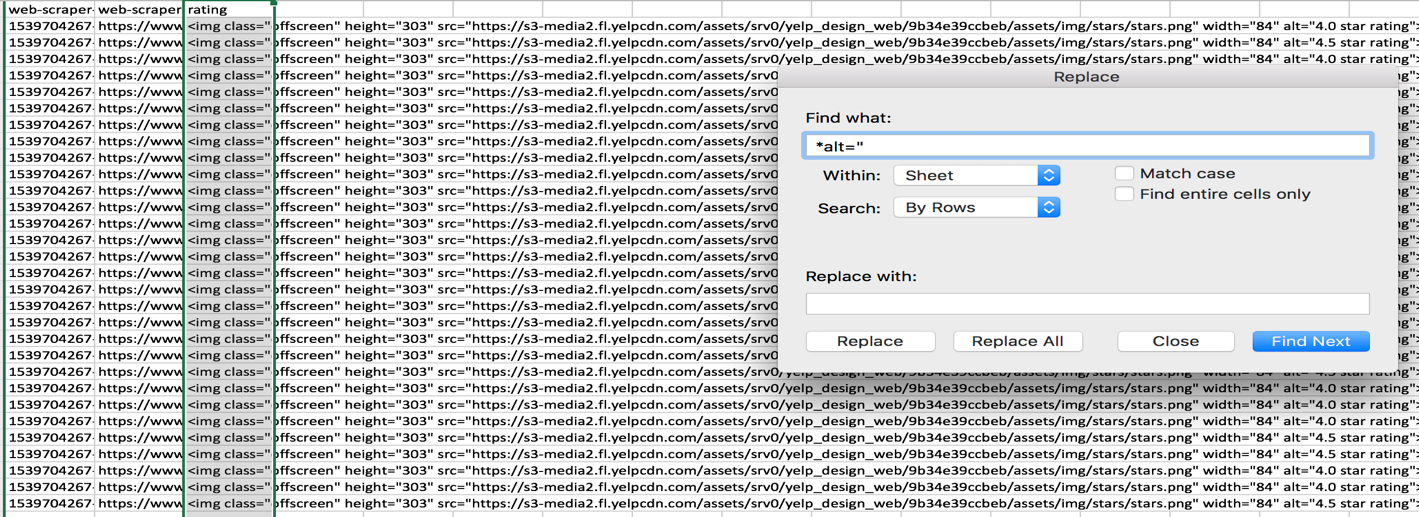
reviews\_df=pd.read\_excel("yelp\_reviews.xlsx",encoding='utf8', errors='ignore')

Again, make sure the review column in your data file is called Restaurant\_review

1. Now run Similarity\_spacy.py. The output file is similarity.xlsx. In this file, sort the rows from high to low similarity scores. Now you can perform sentiment analysis on the 200 reviews with the highest similarity.
2. For task F, you will need the Rating column in the data file. This column will have a lot of html stuff at this point. But what you need is just the rating, e.g., 3.5 or 5.0. Do the following:
3. Select the Ratings column and Press Ctrl+H to get the find the replace window



1. Now enter \*alt=" in Find what: box (make sure no space before or after)



1. Now click on Replace All
2. Again select the ratings column and get the find and replace window and now enter star rating"> in the Find what: box and Click Replace All

