DSCI-560 Assignment No. 4 - Part 1
Instructor: Young Cho, Ph.D.

This assignment is the first part of a 2-part assignment that focuses on providing you with web scraping, data preprocessing, semi-automated topic selection, clustering algorithms, and real-time data processing. You will better understand how to collect and organize data from online forums and create a system for grouping related messages.

You will work with your team in this lab on data collection and topic selection. Additionally, you'll preprocess the data to remove unwanted forum posts and store them in a database.

1) Initial Setup

a) Tools / Libraries

We will be using requests/selenium, beautifulsoup4, mySQL database for this assignment. The installations are similar to the ones used earlier, and you should already have them setup.

Linux / Ubuntu/Python script should be used for the assignment. **(Make sure to document any setup steps / requirements for running your scripts in the document you submit)**

Do not spend much time on the installation and setup; instead, invest your time on exploring the concepts and improvising your submission.

b) Resource

Use the following resource to understand how to scrape Reddit data using BeautifulSoup or the **Praw** API. Both methods have their merits, and your team is free to pick one of these methods based on how you want to structure your web scraper.

Scrape Reddit on python: Scrape Reddit

https://brightdata.com/blog/web-data/how-to-scrape-reddit-python

Praw API: Scraping Reddit using Python Reddit API Wrapper (PRAW)

https://medium.com/analytics-vidhya/scraping-reddit-using-python-reddit-api-wrapper-praw-5c275e34a8f4

BeautifulSoup4 Guide: Scraping Reddit with Python and BeautifulSoup 4

https://www.datacamp.com/tutorial/scraping-reddit-python-scrapy

2) Data Collection / Storage

For this task, you will focus on creating the database tables, fetching posts from this Reddit forum: https://www.reddit.com/r/tech/ using the **Praw** API and writing a python script to populate the database tables from the dataset. Also read about the timeout and max limit of API requests and modify your code so that it can handle large requests.

For example, if API has a threshold of 1000 posts or a timeout of 60 secs. Make sure your code can handle requests of size 5000 or requests that take 400 secs to fetch all results by calling the API multiple times without letting it run out of bounds to fetch all the results correctly and not letting the request fail.

Your script must take the number of posts to fetch as an input and fetch them and store it in the database after preprocessing.

3) Data Preprocessing

Preprocess the data by removing HTML tags, special characters, and irrelevant information such as promoted messages, advertisements, etc.

Transform the data into a suitable format for analysis, such as converting timestamps, masking the user names to maintain data privacy. Identify keywords and topics based on messages and store them as additional fields in the database along with the actual messages.

Some messages might have images embedded in them. Use the image recognition tools like pytesseract and OCR readers used in Assignment 2 to extract text from these images and store them as additional fields in the database. Consider this additional text as well while identifying keywords and topics that the messages belong to.

4) **Resources:**
   a) **Handling Missing Data with Pandas: Dealing with Missing Data in Pandas**

      https://pandas.pydata.org/pandas-docs/stable/user_guide/missing_data.html

   b) **Keyword Extraction: Keyword Extraction process in Python**

      https://towardsdatascience.com/keyword-extraction-process-in-python-with-natural-language-processing-nlp-d769a9069d5c

5) **Team Discussions**

   Your team is expected to meet in-person / virtually each day of the week and discuss the assignment progress & next steps. Document and compile minutes of all meetings in a separate file called **'meeting_notes_A4_P1_<team_name>.pdf'**

6) **Submission**

   Make one submission per team. Each team must submit all the code files for the working solution, a readme document containing information for running the code in pdf format and a document that outlines the minutes of all team meetings in pdf format.

   Provide a video per team which demonstrates the entire working solution and explains how the data tables were loaded, demonstrate query results and talk about the design decisions made along with reasoning for the same. Also include details about how your team preprocessed the data. Please include the team name and the name of all three team members in the video.

   There will be a 50% penalty for all late submissions.