Programming Task: Research Internship at Financial Services Innovation Lab, Georgia Tech

This task is split into multiple parts or steps to make it easier for you to execute in step by step fashion. They are not separate tasks.

- In the end, it is one task and it is due on 10th day from the day you receive the task. But if you need more time to finish please ask for an extension.
- If you will submit the programming task satisfactorily, we will send you a research paper to write a critique on that.
- **IMPORTANT**: Writing correct code is necessary but not sufficient to complete the task. We will also look at how you document the code (README file, docstrings, comments etc.). Follow best coding practices.

Submission Details

- Submit through GitHub. Upload all your work in a private GitHub repository and once you are done with your work share GitHub repository with Agam (GitHub username: shahagam4).
- This task is in Python. You are not allowed to use any other programming language.
- You have to submit ONLY
  - The program used to download the data
  - Note that you do not need to submit any datasets for this task, but please save the data as it will be used in latter parts of this task
- **IMPORTANT**: Please note that this task requires some self-study and research; it is imperative that students begin working on this task early

Task 1: Download Data from the SEC-EDGAR

The main task for the first part of the task is simple. You need to download 10 (random) 8-Ks for each year-quarter (not year) from the SEC website for the time period 1995:Q1 through 2021:Q4 (resulting in approximately $1,080$ 8-K documents). These documents will be analyzed in the next part of the task. Note that the task is to write a script or program to do this automatically, and not to download all files manually for every year-quarter.

- The 8-K files are available on the SEC website: `https://www.sec.gov/`
- More information about accessing SEC data can be found here:
`https://www.sec.gov/edgar/searchedgar/accessing-edgar-data.htm`
- Please be sure to use the *index* file (full-index)
- The directory is organized by year and quarter; thus, if you want to access data from 1995:Q2, you will have to navigate to:
`https://www.sec.gov/Archives/edgar/full-index/1995/QTR2/`
- You can either download the master.idx file or master.zip (preferred)
- Note that you will need to *clean* the master.idx file (after unzipping), in order to remove the first few lines
- Use regular expressions (*regex*) to filter the forms so that you can keep only the 8-K filings
- Extract 10 random lines or companies for each year-quarter
- Extract the path name or link for the 8-K download
- Download the corresponding 8-Ks
- Create a .CSV file that keeps track of the company identifier (CIK) and the 8-K filing date (this will be used for a later part of the task)
- Commit the program (but not the downloaded 8-K files)

Task 2: Rudimentary Sentiment Analysis

- Note: You have to submit ONLY
  - The program used to merge data, clean data, perform sentiment analysis, generate sentiment score and time-series plot.
  - You can also upload the final output (CSV) containing sentiment score for each 8-K file.

The main task for this part is to compute sentiment score for each 8-K filings using very simple bag-of-words approach.

- Please utilize the words list from Bill McDonald's website:
  `https://sraf.nd.edu/textual-analysis/resources/`
  - Please use the latest version of the "Loughran-McDonald Master Dictionary" available
- For each downloaded 8-K filing (from task 1) calculate the difference between the number of positive words and the number of negative words, and scale this difference by the total number of words in the document
- Remember, you need to remove HTML and clean files to reduce noise.
- Generate descriptive statistics for the sentiment measure you calculated for 8-K filings.
- Plot average sentiment score over time. (x-axis: year, y-axis: average sentiment score)