DSCI 551 – HW3 (Spring 2021)

Chat Data Analysis using Python & MySQL

100 points, Due 3/28

Similar to homework 1, we will analyze your Zoom chat logs and also roster data. The sample data are the same as ones provided in homework 1. However, in this homework, you will be using MySQL to store the data. Refer to handout on how to set up python connector for MySQL. We will test your code assuming data are stored in dsci551 database with user dsci551 and password Dsci-551. You need to use MySQL functions to perform searching and analysis whenever you can. Note that your codes may be tested using additional chat and roster data in the same format as the samples.

1. [Data import, 30 points] Write a data importer "import.py" that loads the data into MySQL as two tables, one for chat log, the other for roster. You can design the structure of tables yourself, but do not lose any original information provided.
   Execution format:
   python import.py <chat-log-file> <roster-file>
   For example, python import.py 551-0125.txt 551-mw-roster.csv

2. [Analysis, 30 points]
   a. Write a Python script "stats.py" that computes the total number of chats for each person who participated in the chat. Output the statistics in a JSON file named stats.json. Use data from sql only to solve this question.

      Execution format: python stats.py <output-json-file>
      For example, python stats.py stats.json

      Format of your output file:

      [{"Person":"John Smith","Message":8},…]

   b. Write a Python script "nochats.py" that finds the students who did not have chat messages and their participation locations. Write output also to a JSON file named nochats.json. Use data from sql only to solve this question.

      Execution format: python nochats.py <output-file>
      For example, python nochats.py nochats.json
      Format of your output file:
      [{"Name":"David Chen","Participating from":"United States of America"},…]

3. [Searching with MySQL, 40 points]

a. [20 points] Write a Python script "search-person.py" that finds all students whose name contains at least one of the specified keywords (case insensitive).
For example, python search-person.py 'john smith' will find all students whose name contains either 'john' or 'smith' or both. The keywords might be “john” (length==1) or “john smith” (length==2). We’ll test your code with keywords whose length==1 or length==2. No substring cases need to be considered.

Return the student names one line per student.

Execution format: python search-person.py 'John SMITH'

Example output: (please print the output directly)

John Smith
John Allen
Mary Smith

If no student has the same names as your inputs in this class, please print 'Student Not Found'. No substring cases need to be considered.

b. [20 points] Write a Python script "search-message.py" that finds all messages made by a given student.
For example, python search-message.py 'john smith' will find all chat messages made by a student whose name is 'john smith' (case insensitive).
Output the messages tab separated and one line per message.

Execution format: python search-message.py 'John SMITH'

For example, (please print the output directly)

4:06pm list
5:12pm variety,…
…

If no student has the same name as your input in this class, please print 'Student Not Found.'; if this student is in this class but has no chat messages, please print 'This student is quiet.'

You should use Pandas DataFrame, MySQL fulltext search function , mysql.connector for this homework. Other libraries permitted in this homework are: sys, re, json, and requests.

Scripts that take long run time (e.g. more than 2 minutes) will lead to deductions of points.

Please use Python 3.8 and make sure your scripts can work on ec2 for all the homework. Submission:

a zip file that contains all the above scripts and output files in question 2, with specified names. Name your zip file: John_Smith_hw3.zip.