

DSCI 553: Foundations and Applications of Data Mining

Fall 2020

Assignment 1

Deadline: 9/15/2020 11:59 PM PDT

1. Assignment Overview

In this assignment, you will utilize your Spark knowledge to complete three tasks on the gamergate dataset. You will complete tasks that data scientists and researchers will first do when confronting a new dataset. These tasks will also assess your knowledge of Spark operation-types.

2. Requirements

Please pay attention to these requirements as they will play an important role in your grades!

2.1 Programming Requirements

You must use Python to implement all tasks. You can only use standard python libraries (i.e., external libraries like numpy or pandas are not allowed).

2.2 Programming Environment

Python 3.6, Pyspark 3.0.0

We will use these library versions to compile and test your code. There will be a 20% penalty if we cannot run your code due to the library version inconsistency.

2.3 Write your own code

Do not share code with other students!!

For this assignment to be an effective learning experience, you must write your own code! We emphasize this point because you will be able to find Python implementations of some of the required functions on the web. Please do not look for or at any such code!

TAs will combine all the code we can find from the web (e.g., Github) as well as other students' code from this and other (previous) sections for plagiarism detection. We will report all detected plagiarism.

2.4 What you need to turn in

Your submission must be a zip file with name: **firstname_lastname_hw1.zip** (all lowercase). You need to pack the following files in the zip file (see Figure 1):

- a. three Python scripts, named: (all lowercase)
firstname_lastname_task1.py, firstname_lastname_task2.py, firstname_lastname_task3.py
- b. You don't need to include your results. We will grade on your code with our testing data (data will be in the same format).

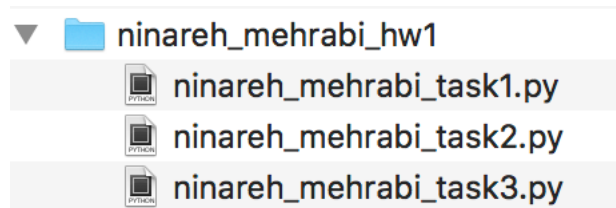


Figure 1: Submission Structure

3. Gamergate Dataset

In this assignment, you will work with two datasets (Gamergate.json and tweets) both provided to you along with this assignment. The Gamergate.json contains metadata for the gamergate twitter dataset which is a data collected from twitter. Each line has information about a specific tweet with different metadata associated. **Note: Each line in the data file contains a tweet in json format.** The tweets data contains all the text tweets from the Gamergate.json in the form of a textfile.

4. Tasks

4.1 Task 1: Extracting General Information from Data

For this task, you will explore the Gamergate.json dataset and gather some general information about this dataset. This is what data scientists and researchers do once they encounter a new dataset. It is important that you first familiarize yourself with the data itself and its structure in general with different fields in it, so open the data and take a look at it before starting the tasks.

For Task 1 you need to answer the following questions:

- A. How many tweets are in this dataset? (5 points)
- B. How many unique users are in this dataset? Hint: each user has its unique user id. (5 points)

- C. Identify top 3 users with most followers. We would need you to output the screen name of these users along with the number of followers they each have. (5 points)
- D. Each tweet has an associated date to when it is created. Identify the number of tweets that are created on Tuesday. (5 points)

Input format: (We will use the following command to execute your code)

```
python firstname_lastname_task1.py <input_file_name> <output_file_name>
```

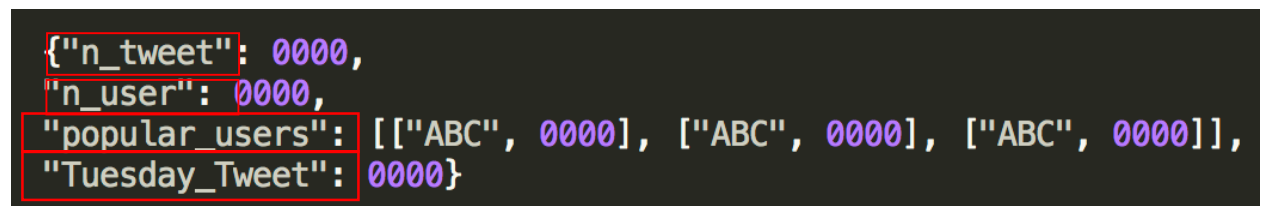
Param: input_file_name: the name of the input file (Gamergate.json), including file path

Param: output_file_name: the name of the output JSON file, including file path

Output format:

IMPORTANT: Please strictly follow the output format since your code will be graded automatically.

- a. The output for Questions A/B/D will be a number. The output for Question C will be a list, which is sorted by the number of followers in the descending order.
- b. You need to write the results in the JSON format file. You must use **exactly the same tags** (see the red boxes in Figure 2) for answering each question.



```
{"n_tweet": 0000,
 "n_user": 0000,
 "popular_users": [["ABC", 0000], ["ABC", 0000], ["ABC", 0000]],
 "Tuesday_Tweet": 0000}
```

Figure 2: JSON output structure for task1.

4.2 Task 2: Statistical Analysis of Data

After researchers get familiar with the data and its general format and information contained, they start gathering some statistical information on the data. In this task, you will perform some basic statistical analysis on the data.

For Task 2 you need to answer the following questions:

- A. Each tweet has a retweet count associated to it. What is the mean retweet count for tweets in this dataset? (on average, how many retweets each tweet gets) (5 points)

B. What is the maximum retweet count? (5 points)

C. What is the standard deviation for the retweet counts? (5 points)

Input format: (We will use the following command to execute your code)

```
python firstname_lastname_task2.py <input_file_name> <output_file_name>|
```

Param: input_file_name: the name of the input file (Gamergate.json), including file path

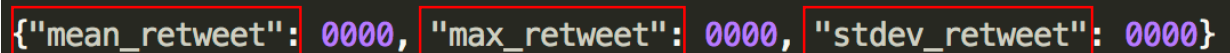
Param: output_file_name: the name of the output JSON file, including file path

Output format:

IMPORTANT: Please strictly follow the output format since your code will be graded automatically.

a. The output for Questions A/B/C will be a number. **Note: Do Not round the numbers!**

b. You need to write the results in the JSON format file. You must use **exactly the same tags** (see the red boxes in Figure 3) for answering each question.



```
{"mean_retweet": 0000, "max_retweet": 0000, "stdev_retweet": 0000}
```

Figure 3: JSON output structure for task2.

4.3 Task 3: Text Processing

Finally, since our data is a twitter dataset one of its most important components is the text or tweets written by the users. Thus, it is important to perform text processing and analysis on this. In the research pipeline, this is what researchers pursue. For this task, you will use the **tweets** file (provided to you as part of the assignment) which is a file containing text from tweets in the gamergate dataset.

For Task 3 you need to answer the following questions:

A. What is the most frequent word in this file with what frequency? (5 points)

B. How many times the word “mindless” appears in the file? (5 points)

C. Each tweet chunk is between |*****| chunk
*****| identifiers. how many tweet chunks are there in this
file? (5 points)

Input format: (We will use the following command to execute your code)

```
python firstname_lastname_task3.py <input_file_name> <output_file_name>
```

Param: input_file_name: the name of the input file (tweets), including file path

Param: output_file_name: the name of the output JSON file, including file path

Output format:

IMPORTANT: Please strictly follow the output format since your code will be graded automatically.

- The output for Questions B/C will be a number. The output for Question A would be the most frequent word followed by its frequency as shown in Figure 4.
- You need to write the results in the JSON format file. You must use **exactly the same tags** (see the red boxes in Figure 4) for answering each question.

```
{"max_word": ["ABC", 0000], "mindless_count": 0000, "chunk_count": 0000}
```

Figure 4: JSON output structure for task3.

5. Grading Criteria

Perfect score for this assignment is 50 points.

Assignment Submission Policy

Homework assignments are due at 11:59 pm on the due date and should be submitted in Blackboard. Every student has **FIVE free late days** for the homework assignments. You can use these five days for any reason separately or together to avoid the late penalty. There will be no other extensions for any reason. You cannot use the free late days after the last day of the class. You can submit homework up to one week late, but you will **lose 20% of the possible points** for the assignment. After one week, the assignment cannot be submitted.

(% penalty = % penalty of possible points you get)

- You can use your free 5-day extension separately or together.
- If we cannot run your programs with the command we specified, there will be 80% penalty.
- If your program cannot run with the required Python/Spark versions, there will be 20% penalty.
- If our grading program cannot find a specified tag, there will be no point for this question.
- If the outputs of your program are unsorted or partially sorted, there will be 50% penalty.
- If the header of the output file is missing, there will be 10% penalty.
- We can regrade on your assignments within seven days once the scores are released. No argue after one week. There will be 20% penalty if our grading is correct.
- There will be 20% penalty for late submission within a week and no point after a week.
- There will be no point if the total execution time exceeds 15 minutes.