COMP 598 Homework 3 – Unix and python analysis

30 pts
Assigned Sept 23, 2021
Due Oct 1, 2021 @ 11:59 PM

The goal of this assignment is to start using some of our standard tools to do some core data science work.

Task 1: Data filtering (15 pts)

Write a bash shell script stats.sh that accepts as a command line argument a tweet file (similar to the one in homework 1) and prints out, on subsequent lines:

- The number of lines in the file
- The first line of the file (i.e., the header row)
- The number of lines in the last 10,000 rows of the file that contain the string “potus” (case-insensitive).
- Of rows 100 – 200 (inclusive), how many of them that contain the word “fake”

All this should be done only using standard Unix commands and pipes.

To be clear, your script should work on any file which has at least 10,000 lines in it. Print an error message if the input file is smaller than 10,000 lines.

It will be tested by TAs by calling it using:

stats.sh <test_file>

And it should output something like:

```
~$ sh stats.sh sample.csv

1000001

col_1, col_2, col_3, col_4

1234

56
```

The TAs will get to choose the test file.

Task 2: Watch some My Little Pony episodes (0 pts – totally optional)

In this and the next homework, we’re going to be analyzing My Little Pony language. As we’ve discussed, it’s always important to study your source material … particularly when it’s very entertaining cartoons! So if you’re able, watch a couple episodes!

Task 3: Basic My Little Pony Talking Stats (15 pts)

We’ll be using the dataset available here: https://www.kaggle.com/liury123/my-little-pony-transcript

For the purpose of this study, we’ll use only clean_dialog.csv and assume that the dataset is perfect.

Write a python script named **dialog_analysis.py** that, when run, computes and produces a JSON-formatted result that has exactly the structure given below (all numbers below are just examples).

```
{
```

```
  "count": {
    "twilight sparkle": 327,
    "applejack": 42,
    "rarity": 332,
    "pinkie pie": 123,
    "rainbow dash": 412,
    "fluttershy": 125
  },
  "verbosity": {
    "twilight sparkle": 0.37,
    "applejack": 0.24,
    "rarity": 0.09,
    "pinkie pie": 0.10,
    "rainbow dash": 0.10,
    "fluttershy": 0.10
  }
}
```


Notice this JSON file has two keys: count shows the number of speech acts that each character has in the entire file. verbosity gives fraction of dialogue, measured in # of speech acts produced by this pony. For both cases, we only want to see the values related to the six ponies (the main characters of the cartoon). Any other character should not be present in this exercise.

Important:

- Please make sure your JSON file respects this structure and watch out for matching the keys accordingly.
- A pony is only considered the speaker if their name is a COMPLETE match with their name (case-insensitive to be a bit forgiving). So if the speaker is “Twilight”, this is NOT attributed to “Twilight Sparkle”. Whereas “rainbow dash” IS attributed to “Rainbow Dash”.

Your script should run as follows:

```
python3 dialog_analysis.py -o output.json clean_dialog.csv
```

Your analysis script should use only standard python libraries and pandas if needed (note it's optional – you don’t have to use pandas for this). Output the results in a JSON file named **output.json**. Note that the

“clean_dialog.csv” and “output.json” are arguments for your analysis script. Therefore, if the TAs change the name of the clean_dialog.csv file to clean_dialog_grading.csv, then they would run it as

```
python3 dialog_analysis.py -o output.json clean_dialog_grading.csv
```

Submission Instructions

- Check the README.md file on the Github repository for HW3.

We will run your dialog_analysis.py script and regenerate the output.json file.

Hints

How to solve this assignment? Here are some important hints:

1. Download and inspect the dataset first. The questions will make much more sense once you see how the data looks like.
2. How do you calculate the verbosity? Imagine there are two elements of interest, A and B, and there are 10 acts. A speaks 7, B speaks 3; so the percentages are straightforward: A: 0.7; B: 0.3. But remember you might encounter other elements in the dataset that you are not interested in - but they should be used when computing the percentage because we are looking at the number of speech acts.