COMP 598 Homework 5 – Data Collection & Cleaning

30 pts
Assigned Oct 19, 2020
Due Oct 29, 2020 @ 11:59 PM

This is an INDIVIDUAL Assignment – each student's work must be their own, each student completes this assignment, there are no teams for homework 5.

Non-standard (i.e., built-in) python libraries you can use:

- pandas
- requests

Cleaning data (10 pts)

You are cleaning files containing user posts in JSON data. The JSON objects are all valid, but they have field consistency issues that you need to fix. Your script, clean.py, should accept an input file and produce a cleaned output file. The script is run:

```
python3 clean.py -i <input_file> -o <output_file>
```

The input file will contain one JSON dictionary per line (see the file example.json included with the assignment which you can use as an example… though you shouldn't treat it as a complete test. Your script will need to handle arbitrary JSON dictionaries). Each dictionary is a post. The output file should also have one JSON dictionary (i.e., post) per line. Your script should do the following:

- Remove all the posts that don't have a *title* or *title_text* field.
- For objects with a "title_text" field, it should be renamed in the output object to "title"
- Standardize all *createdAt* date times to the UTC timezone. Any post that has an invalid date time that can't be parsed using the ISO datetime standard should be removed.
- Posts that haven't been flagged for removal should be written to the output file in the order they appear in the input file.
- Any lines that contain invalid JSON dictionaries should be ignored.

Tips:

- Python provides a very powerful datetime package called… datetime
- When working with JSON data, the json package is your friend.

Data Collection & Bias (20 pts)

You've been given the task of assessing the average length of a Reddit post title. But since we can't collect ALL Reddit posts, we're going to try two different ways of sampling posts to assess their length:

- Sample 1: Collect the 1000 newest posts from the 10 most popular subreddits by subscribers: funny, AskReddit, gaming, aww, pics, Music, science, worldnews, videos, todayilearned.
- Sample 2: Collect the 1000 newest posts from the 10 most popular subreddits by # of posts by day: AskReddit, memes, politics, nfl, nba, wallstreetbets, teenagers, PublicFreakout, leagueoflegends, unpopularopinion

Compute average post title length (measured as # of characters/post text) for each scenario.

This task should be split into two scripts: collect.py and compute_title_lengths.py

collect.py does the work of collecting the data for BOTH of the sampling approaches. It should store the data (as received from Reddit) in files data/sample1.json and data/sample2.json (respectively). It can work however you like – we will not grade it directly.

compute_title_lengths.py should accept an input file containing one JSON dict per line (corresponding to a subreddit post) and output the average post title length – in essence, it should accept one of the sample.json files producted by the collect.py script. The input JSON dict should respect *exactly* the format returned by reddit's API. The script is called like: python3 compute_title_lengths.py <input_file>

Important aside: Once you've implemented the scripts and computed the average title lengths, reflect for a moment about how these two quite reasonable definitions of random posts have produced different results (this is ungraded – but an important part of this task). In a real project, you'll have to pick and be prepared to defend only one approach. This is the challenge that data scientists run into all the time. Drop in for office hours and let me know which you'd pick and why. Or is there another approach that you think would be better? I'm eager to discuss and debate.

Tips:

- If the post title doesn't contain any text, it still counts as a post … just with text length of zero.
- Note that you are collecting POSTS, not comments (which are the responses to posts).
- To collect the posts you'll want to use the /new API endpoint.
- make sure to set the User-Agent in your get requests … see API guidelines here: https://github.com/reddit-archive/reddit/wiki/API.
- In order to get this data, you may need to authenticate to Reddit. Instructions here: https://github.com/reddit-archive/reddit/wiki/OAuth2
- The packages requests + json will be your friends.

Submission Instructions

Your MyCourses submission must be a single zip file entiled HW5_<studentid>.zip. It should contain the following items:

- scripts/
  - clean.py – the script for Task 1
  - collect.py – the script for Task 2
  - compute_title_lengths.py – the script for Task 2
- src/
  - hw5/ - this directory will be included in the PYTHONPATH when clean.py and compute_title_lengths.py are run
    - <whatever you want here>…
- data/
  - sample1.json – the 1000 posts you collected from the first set of subreddits
  - sample2.json – the 1000 posts you collected from the second set of subreddits