CS6501-009 : Homework
Due: 1st October, 2019

Goals:

Gain hands-on experience with the MapReduce framework. MapReduce, initially developed by Google (see paper below), is a programming paradigm used primarily in Hadoop systems. MapReduce provides a more efficient way to process a large amount of data. The task is to sort the contents of a large data file (book).

This assignment will have two parts. The first is to sort the words and do some tasks on a larger data file (a book). Lastly, we will look at more complex MapReduce programs as well as an introduction in Amazon's Elastic MapReduce .

For this assignment we will use Python and the built-in 'mrjob' library.

Link to original Google paper:
https://www.usenix.org/legacy/events/osdi04/tech/full_papers/dean/dean.pdf

Step 0: Create an AWS account

1. Create a free account on https://aws.amazon.com/.
2. Apply for AWS Educate here https://aws.amazon.com/education/awseducate/apply/ -- free $40 credit for students. Select the option "Students". Note that YOU will be responsible for paying for AWS! But $40 should be enough with careful usage for all the programming assignments.

Please apply ASAP! Normally, it will take more than one day for the application to get approved.

Step 1: (In each of the following steps, you need to refer to the instructions in this step)

Follow the instructions in the official 'mrjob' tutorial:
https://mrjob.readthedocs.io/en/latest/guides/quickstart.html

If you have never used Python before, you may want to refer to https://www.youtube.com/watch?v=YYXdXT2l-Gg (Windows and Mac) or https://www.youtube.com/watch?v=UXjjEZroOu0 (Linux). You can find a lot of beginning tutorials online.

Step 2:

Download the text for *Harry Potter and the Prisoner of Azkaban* here: http://prometheusufo5.tripod.com/ebooks/HarryPotter3-PrisonerAzkaban.txt

Calculate the word frequency for the text file. The word frequency is defined by the number of times a word appears in the book. Then sort the words in the ascending order of frequency. Specifically, please find the number of occurrences for the following words: ‘magical’, ‘soaring’, and ‘lopsided’ and include the results in your final submission.

You may have noticed that the output list of word frequencies includes the same word with additional punctuation (e.g., "late", "late,", "late.\"). Please refer to the section “Writing your second job” in the ‘mrjob’ tutorial above or search “python remove punctuation” in Google to solve this problem and only show the word without punctuation in the output.

Step 3:

Using the structure from step 2, you can find out the frequencies of the words in *Harry Potter and the Prisoner of Azkaban*. Please list the top ten most frequently used words and their associated word count in the final results.

Hint: You may have noticed that to find the top ten most frequently used words, the previous issue in step 2 with punctuation must be fixed. You may also want to use more than one mapper/reducer (the additional mapper/reducer is for sorting) or more than one program (the second program prints out the final result). Please refer to https://mrjob.readthedocs.io/en/latest/guides/quickstart.html#writing-your-second-job for the multi mapper/reducer.

Step 4:

The last step of this homework is to get basic experience using Amazon Elastic MapReduce (EMR). To do this part of the assignment, please follow the following steps:

1. Configuring AWS credentials

Configuring your AWS credentials allows mrjob to run your jobs on Elastic MapReduce and use S3.

- Create an Amazon Web Services account
- Go to Security Credentials in the login menu (upper right, third from the right), say yes, you want to proceed, click on Access Keys, and then Create New Access Key. Make sure to copy the secret access key, as there is no way to recover it after creation.

Now you set aws_access_key_id and aws_secret_access_key in your mrjob.conf file like this:

```
runners:
  emr:
    aws_access_key_id: <your key ID>
    aws_secret_access_key: <your secret>
```

2. Run your job with -r emr
You can store this file at /etc/mrjob.conf, ~/.mrjob.conf, or ./mrjob.conf. Then run your job with -r emr:

```
python your_mr_job_sub_class.py -r emr < input >
```

E.g., python your_mr_job_sub_class.py -r emr harry.txt

If you store this file at a different directory, to run your job with -r emr, you need to pass it via --conf-path using the command below.

```
python your_mr_job_sub_class.py -r emr --conf-path <YourDirectory> < input >
```

E.g, python step4.py -r emr --conf-path /af1/hs6ms/mrjob.conf harry.txt

(Optional) For more details, you can refer to https://mrjob.readthedocs.io/en/latest/guides/emr-quickstart.html .

So far, you have run your script from step 3 in EMR. Please verify that the output matches your results from step 3 and include screenshots of your console log (this is the output in command prompt or terminal).

A few things to note:

- When running the script, you need to replace <input> with the input data file such as harry.txt.
- Feel free to use an existing security key that you already have for your 'mrjob.conf' file.
- You may get the following message in terminal: "Waiting 10 minutes for logs to transfer to S3... (ctrl-c to skip)". If you get this, please 'ctrl-c'
- This portion will take time as Amazon has to create and then terminate a cluster. If you find yourself waiting for ~15 minutes, do not worry!

Submission:

Please include ONE PDF containing the results mentioned in this documentation and description of your solution approach.

Important notes:

- All these programs (especially in the steps 2) can be done iteratively without using MapReduce. However, to receive credits on these parts, you must perform your calculations using MapReduce.
- Please start this assignment several days in advance of the deadline. This is largely due to the number of programs being written (though most follow a similar general structure)
- When running these programs, include the name of the data file (e.g. 'harry_potter.txt') in the command line argument