DAT470/DIT065
Computational techniques for large-scale data

Assignment 4
Deadline: 2024-05-06 23:59

Problem 1: Twitter followers (12 pts)

We will revisit last week's assignment [1] and reimplement it in Spark. As before, the data is found in files `/data/2024-DAT470-DIT065/twitter-2010_*.txt`. Use the file `/data/2024-DAT470-DIT065/twitter-2010_10M.txt` when measuring the scalability of your code.

(a) Implement the missing bits of `pyspark_twitter_follows.py` to determine the maximum number of people followed, the Twitter id of the account with the maximum number of people followed, the average number of people followed, and the number of accounts that follow no-one. (4 points)

(b) Measure the scalability of your algorithm on 1, 2, 4, . . . , 32 cores. Plot the empirical speedup as the function of cores. In addition to the plot, report the single-core runtime on the dataset. (2 points)

(c) Implement the missing bits of `pyspark_twitter_followers.py` to determine the maximum number of followers, the Twitter id of the account with the maximum number of followers, the average number of followers, and the number of accounts that have no followers. (4 points)

(d) Measure the scalability of your algorithm on 1, 2, 4, . . . , 32 cores. Plot the empirical speedup as the function of cores. In addition to the plot, report the single-core runtime on the dataset. (2 points)

Problem 2: OpenSSH logs (12 pts)

The dataset `/data/2024-DAT470-DIT065/SSH.log` [2], originally downloaded from[1], contains 28 days' worth of log information of an OpenSSH server. Your task is to do some analysis of the log files. In particular, we are interested in lines that relate to password authentication. Relevant lines may look like following:

- `Jan 7 17:13:12 LabSZ sshd[30238]: Accepted password for jmzhu from 137.189.204.220 port 52712 ssh2`
- `Jan 7 16:20:02 LabSZ sshd[30105]: Failed password for ftp from 185.222.209.151 port 52023 ssh2`
- `Jan 7 16:20:18 LabSZ sshd[30107]: Failed password for invalid user monitor from 185.222.209.151 port 52863 ssh2`
- `Dec 20 17:25:58 LabSZ sshd[15992]: message repeated 2 times: [ Failed password for curi from 123.255.103.142 port 35777 ssh2]`

[1] `https://github.com/logpai/loghub/tree/master/OpenSSH`

You will need to process the data in such a way that you capture all these cases: an attempt may have been made to log in with a user account that does not exist, and in some cases a large number of repeated messages have been conflated into a single line which you will need to expand.

Write *one* Python file that answers all of these questions, using PySpark, and in addition write down the answers in your report. This time there is no skeleton file, you will have to come up with your own file from scratch. As such, the output of the file is of lesser importance, as long as it produces the correct answers.

(a) Determine which user account had the largest number of login attempts (successful or unsuccessful). Which account was it and how many attempts were there? (2 pt)

(b) Determine which user account had the largest number of *successful* login attempts. Which account was it and how many attempts were there? (1 pt)

(c) Determine which user account had the largest number of *unsuccessful* login attempts. Which account was it and how many attempts were there? (1 pt)

(d) Which user account had the highest *success rate* of logins (successful login attempts / all login attempts). What was the rate? If there were multiple accounts with the same rate, report them all. (2 pt)

(e) Determine the top 3 IP addresses with the largest number of failed login attempts and report the IP addresses together with the number of login failures. (2 pt)

(f) Determine the top 10 invalid user accounts with the largest number of failed login attempts and report the user accounts together with the number of login failures. (2 pt)

(g) Which day had the most login activity? Which day had the least? (2 pts)

Hints

- Obviously, you should get the same results as with the code you used before. This is a good *lithmus test* for the correctness of your solution.
- Using `cache()` correctly can have a drastic effect on performance.
- If you are new to Spark, keep the cheat sheet at hand.
- The `flatMap()` function is very useful if you need to emit multiple values.

Returning your assignment

Return your assignment on Canvas. Your submission should consist of a report that answers all questions as PDF file (preferably typeset in LaTeX) called `assignment4.pdf`. In addition, you should provide the code you used in Problems 1a, 1c, and 2 as `assignment4_problem1a.py`, `assignment4_problem1c.py` and `assignment4_problem2.py`, respectively. The code for problems 1a and 1c must match the interfaces of `pyspark_twitter_follows.py` and `pyspark_twitter_followers.py`; the command line parameters must not be changed, and output must be correct. Do *not* deviate from the requested filenames and do *not* produce the plots in these files; these files will be used for evaluating the quality of your implementations automatically.

References

[1] Haewoon Kwak et al. "What is Twitter, a Social Network or a News Media?" In: *Proceedings of the 19th international conference on World wide web (WWW '10)*. 2010. DOI: `https://doi.org/10.1145/1772690.1772751`.

[2] Jieming Zhu et al. "Loghub: A Large Collection of System Log Datasets for AI-driven Log Analytics". In: *Proceedings of the 34th IEEE International Symposium on Software Reliability Engineering (ISSRE 2023)*. 2023. DOI: `https://doi.org/10.1109/ISSRE59848.2023.00071`.