CS425 MP1 Group 38

Design

In our design, each process starts by reading in a statically configured list of member addresses and checking for local log files. After starting a local RPC server that listens for incoming requests, the process goes into a loop that prompts users for grep commands. Upon receiving a user input, the client process either dials up remote servers or reuses previously established connections to send the command to all members in the list (including the local server). The receiving processes execute the grep command on local log files and send the results back to the client. The client process eventually aggregates all received results and prints them to the command line.

Test

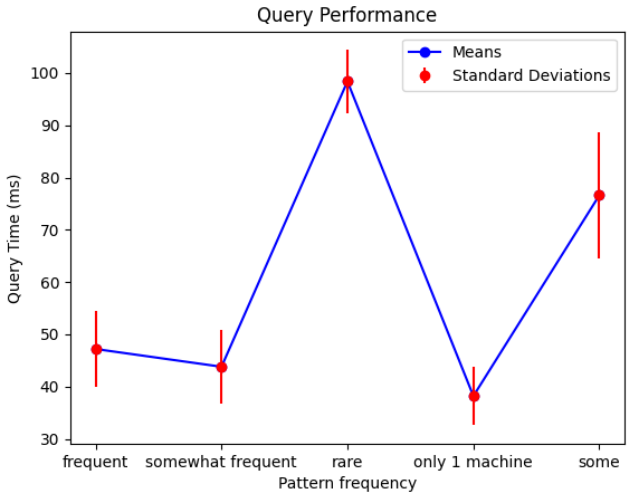
Each unit test will generate a test log file on each machine. The machine that is running the test cases will also have a copy of all the files generated on all machines. Then, a distributed grep is performed through rpc calls (the same function/procedure as the main program). The result of the distributed grep will be compared with a grep command executed on the local copies of the generated log files to determine correctness. Each line in the test log file is a random string generated with characters a-z (both lowercase and uppercase). A number pattern is then inserted based on a predefined probability. In this way, we can precisely control the frequency of patterns.

There are 8 test cases (excluding ones for utility functions): 1. rare pattern (0.1 probability / line), 2. frequent pattern (0.8 probability / line), 3. somewhat frequent pattern (0.5 / line), 4. pattern on some machines (0.7 probability / machine), 5. pattern on one machine, 6. grep with an additional flag (-i), 7. a regular expression pattern, 8. a pattern that contains space and is wrapped with double quotes in input.

Query Performance

Test is performed using VM1~4, each loaded with respective log files distributed on Piazza.

- Frequent pattern: grep -c a
- Somewhat frequent pattern: grep -c log
- Rare pattern: grep -c aa
- Pattern existing on only one machine (vm1): grep -c \[17/Aug/2022:18:29:02 -0500\]
- Pattern existing on some machine (vm1, vm2): grep -c 2022:18:29:02



An observation here is that query time does not seem to increase with frequency and its distribution across machines. During our trials, the first query usually takes longer than the following ones as time used for establishing tcp connection is also taken into account.