

Machine Programming 1 – Distributed Log Querier

An Phan(anphan2), Chang Liu(changl25)

Design

For this project, we adopted a traditional design approach where each machine hosts a server running continuously. These clients maintain lists of IP addresses for all servers in the network and send queries to them independently. When a query is sent to a server, the server responds with the results of the grep command and the total number of matching lines of the host machine. The client receives the responds from all servers and accumulates the total number of matching lines and displays the result on the terminal.

Unit Tests

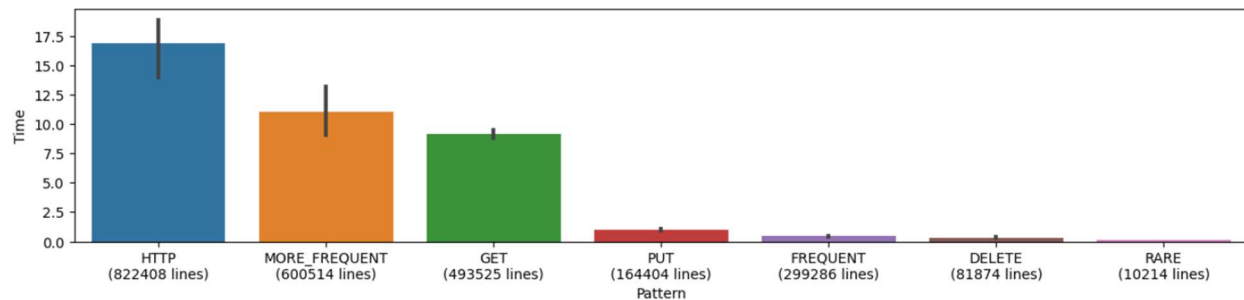
To test whether the `server.py` and `client.py` can run the `grep` command normally and send the result from the servers to client, we generalize the log file of different pattern to test it. We generate ten log files with some known patterns, which are different in the ten log files. For example:

`'VM_ONE'`: only appears in VM1, `'EVEN_FREQUENT'`: only appears in VMs that numbers are even, `'FREQUENT'`: appears in all VMs

We tested each different patterns and compared the results of all VMs with the result directly run `'grep'`.

Experimental Results:

We test the algorithm on different patterns and record the results.



Discussion

As we can see from the plot, the more FREQUENT the pattern is the longer it takes for the process to be finished. We think this is because the FREQUENT pattern produces larger files to send back to the client, therefore slowing down the process. The standard deviation also increases as the pattern becomes more FREQUENT. We think the reason for the rise in standard deviation is because as the responds get bigger, there is more room for error when sending back the responds to the client and the time taken for the process vary more widely.