Team 55 (yixin10, yangt2) – MP 4 Report

1. Design

1.1 Overall Algorithm Design

In our system, VM 1 is designated as the Coordinator. To carry out inference task, the client can set batch size, and input inference command, which includes the input file name in SDFS and its model type. Then the inference request is sent to Coordinator. Upon receiving the request, Coordinator will start a new thread to process it. Within this thread, Coordinator will allocate inference tasks to available machines using **round-robin algorithm**. Specifically, all files this thread iterates will be divided into multiple smaller datasets (each is of batch size). When a worker failed, master will re-calculate the resources allocate to each job according to their estimated inference time and the number of machines running on each job to balance the resources. Tasks that are lost during the failure will be handled before the whole job finished. When master receives inference results from other machines, it will dispatch these results and the query Id to standby backups, and these standby machines will also record them.
When master fails, we used the same master election algorithm implemented in MP3 to select a new master from those backup coordinators, and the new master then will discard some running inference that is allocated by the dead master and the inference results that being transferred to master, and it will re-allocate these tasks and the remaining inference tasks to currently alive workers, and because the new master has been recording previous query Id and the query results, it will not re-query the task that have been completed. Finally, before the job is completed finished, another thread will be started to check if all the input files have been processed. And if there are files that are incorrectly processed, it will be processed again here. Then after the job is completely finished, master will put this inference result file into SDFS.

2. Measurement

(1a) The ratio of resources that IDunno decides on across jobs

When a job is added into system, the system will dynamically adjust the resources allocated to each job according to batch size and the estimated inference time. Specifically, a ratio that represents the amounts of resources utilized between two jobs will be maintained and the task scheduling is based on this ratio, so that fair-time inference can be achieved.
The time interval here is 1 seconds, and it can be observed that in most cases the result is 5:5, but only in one measurement, the result is 4:6. The plot is shown as below in Figure 1.

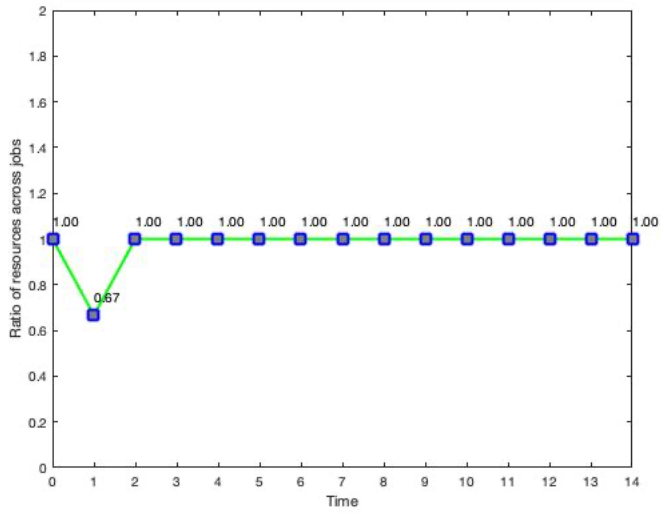

Figure 1: The ratio of resources allocated across on two jobs

(1b) Time spent when cluster starts to inference a new job

We measured the time from when clients inputs the inference command to master starting dispatching the query message, and T_{avg} = 311.3 ms, T_{std} = 140.1 ms.

the plot is shown as below in Figure 2.

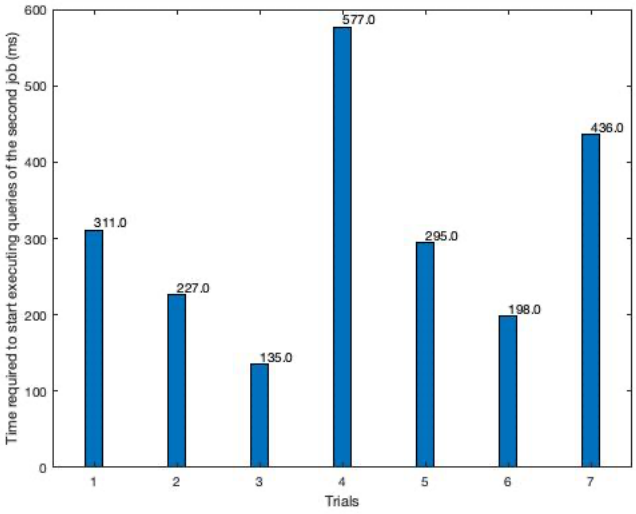

Figure 2: Time spent for cluster to start querying when a job added

(2) Time to resume "normal" when a non-coordinator machine fails

We measured the time the cluster needs to resume when a non-coordinator fails. T_{avg} = 301.1 ms, T_{std} = 77.1 ms. In this system, master will iterate over membership list to find available and alive machine to allocate a job query, so the time is the same as the time our system required to detect the failure of this machine, and remove it from membership list.

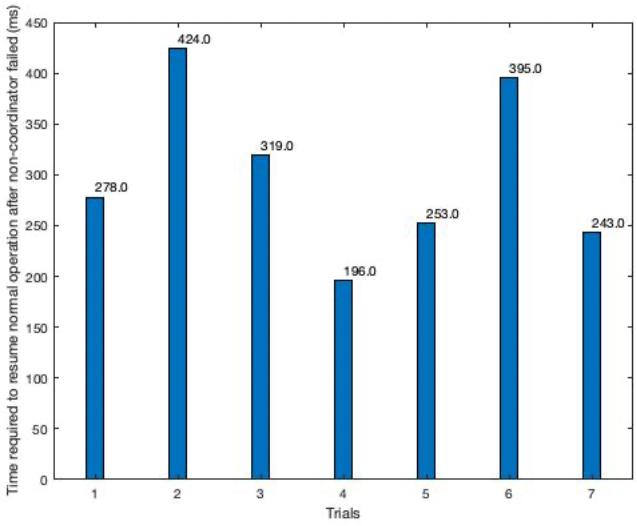

Figure 3: Time to resume when a non-coordinator fails

(3) Time to resume "normal" when coordinator fails

We measured the time the cluster needs to resume when the master fails. T_{avg} = 542.3 ms, T_{std} = 88.2 ms. So compared to the time the cluster needs to resume when a non-coordinator machine fails, a master failure will spend more time to recover. This is because upon detecting master failure, the client will resend its request to back up master, and back up master will send message to other alive machines to ask them to discard some running inference results disseminated by former failed master, which may take more time to process.

The plot is shown as below in Figure 4.

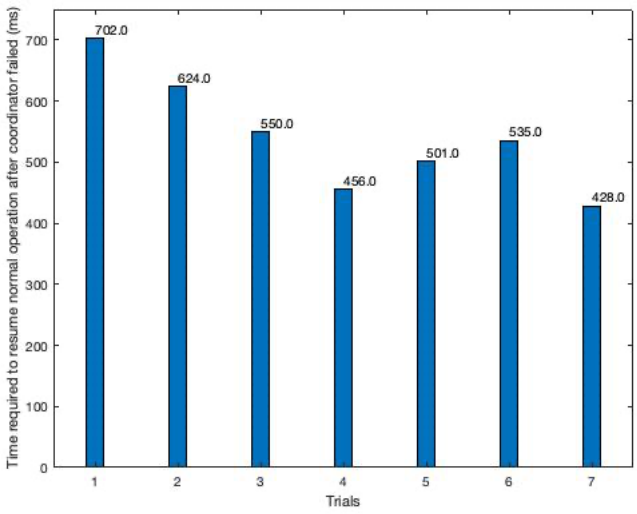

Figure 4: Time to resume when master fails