DSC 102
Systems for Scalable Analytics

Arun Kumar

Review 1

Time tip: Roughly 45sec to 1min per 1pt
Q1) Suppose you magically have a base-3 computer. Consider this custom float6 representation using base-3 digits:

![Float6 Representation](image)

Where the sign is 0, the exponent is 1110, and the fraction is 012.

\[ (-1)^{\text{sign}} \times 3^{\text{exponent} - 4} \times (1 + \sum_{i=1}^{3} d_{3-i}3^{-i}) \]

Q3) [2 x 6pts] What is the decimal real number value (round to 3 decimal places) of these float6 bit sequences interpreted as above?

A. 011012
B. 120210
Q2) Consider this task graph with known task runtimes but all edges between tasks being hidden/unknown.

A. [3pts] Suppose the degree of parallelism is 6. Propose a plausible set of edges that conforms to this.

B. [4pts] Suppose the degree of parallelism is only 4. Propose a new plausible set of edges for this.

C. [5pts] For your task graph in B, what is the highest possible speedup on a cluster with task parallelism vs using just 1 worker with no idle times?
Q3) Suppose you need to run following classification workflow. Try two alternative feature engineering approaches; on each feature set, try both a Naive Bayes model and a Random Forest model. For the latter model, you try 4 different hyperparameter combinations each.

A. [6pts] Draw the task graph for this whole workflow. Make sure to identify/label every task clearly.

B. [2pts] How many workers are necessary to achieve lowest-possible completion time with task parallelism no matter the tasks’ runtimes?

C. [4pts] Suppose the task runtimes are known: 20 and 30 units for the feat. eng. approaches; 5 units to build Naive Bayes on either feature set; 15 to 25 units to build Random Forest depending on the feature set and hyperparameter combination. What is the upper bound on the lowest-possible completion time for this workflow?

D. [2pts] What is the speedup when hitting the upper bound vs using just 1 worker with no idle times for your whole workflow?
Q4) Suppose you have tasks T1-T5 arriving at times 0, 0, 10, 10, and 20, respectively. Suppose Dask gave you the following Gantt chart for running this workload on a cluster of 3 workers.

A. **[6pts]** What is the average turnaround time?

B. **[2pts]** What is the speedup obtained on this cluster vs using just 1 worker with no idle times?

C. **[2pts]** Is it possible to reduce completion time further for this workload by giving Dask additional workers?