Intelligent Resource Provisioning and Allocation Strategies in High Performance Computing

Beste Oztop[1], Benjamin Schwaller[2], Vitus J. Leung[2], Jim Brandt[2], Brian Kulis[1], Manuel Egele[1], Ayse K. Coskun[1]

[1]Electrical and Computer Engineering Department, Boston University, Boston, MA, 02215

[2]Sandia National Laboratories, Albuquerque, NM, 87123

Introduction

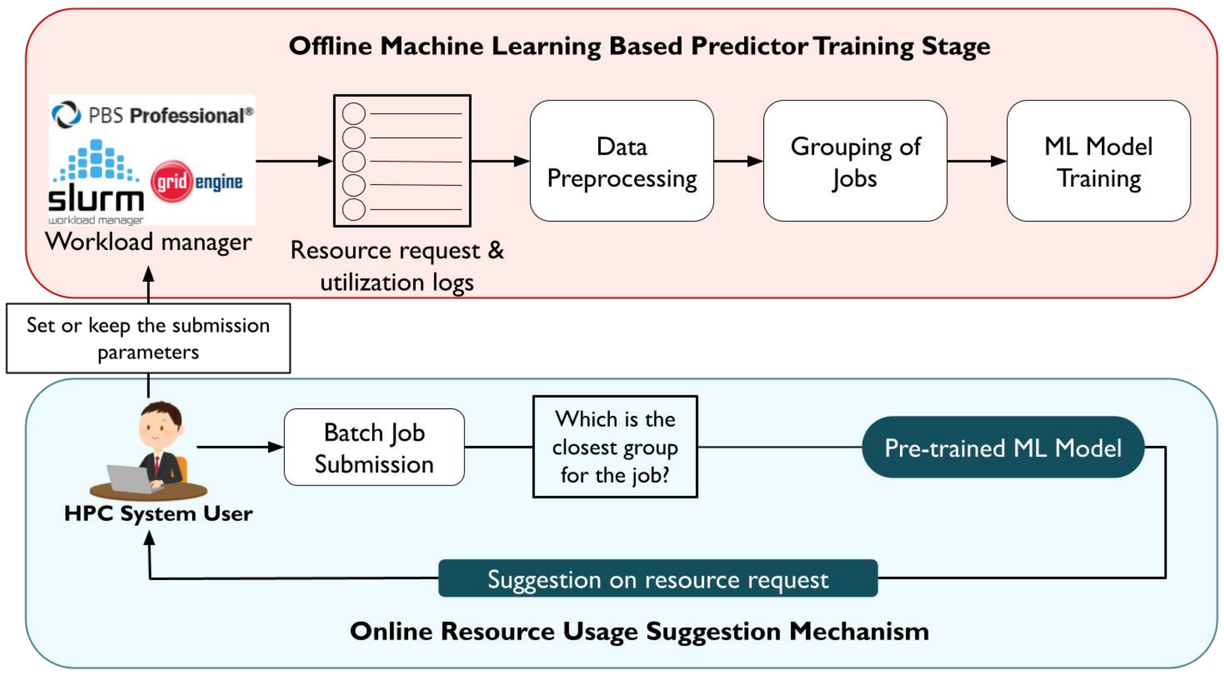
Users **manually** request
- Execution Time
- Number of Processors
- Maximum Memory Size
- Computing Nodes

Underestimation
- Early job terminations
- Reduced system efficiency

Overestimation
- Increased job waiting time
- Idle resources

- Reduce the resource waste and increase the QoS in HPC systems
- Address both under and over requests in batch job submissions

Objectives



Providing online suggestions to the HPC system users after predicting the execution time, maximum memory size and number of processors requirements of batch job submissions

↓ User Wait Time ↑ Job Throughput ↓ CPU Node Hours

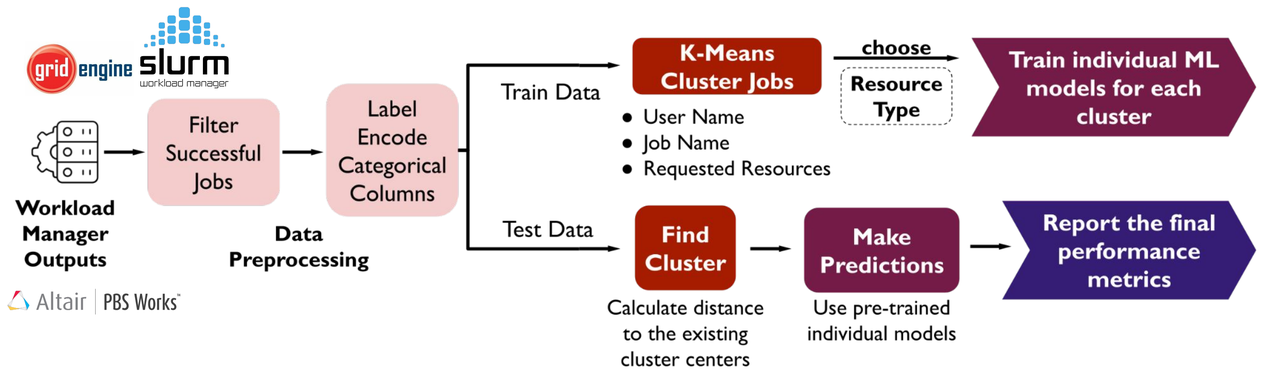
Framework Design

- Group similar jobs according to the user, job, and account names along with resource requests and wallclock time limits
- Apply data driven strategies after model training to prevent the underestimation of resources

Resampling
Changing the training dataset distribution

Buffering
Adding an offset (2σ) to model predictions

Experimental Methodology



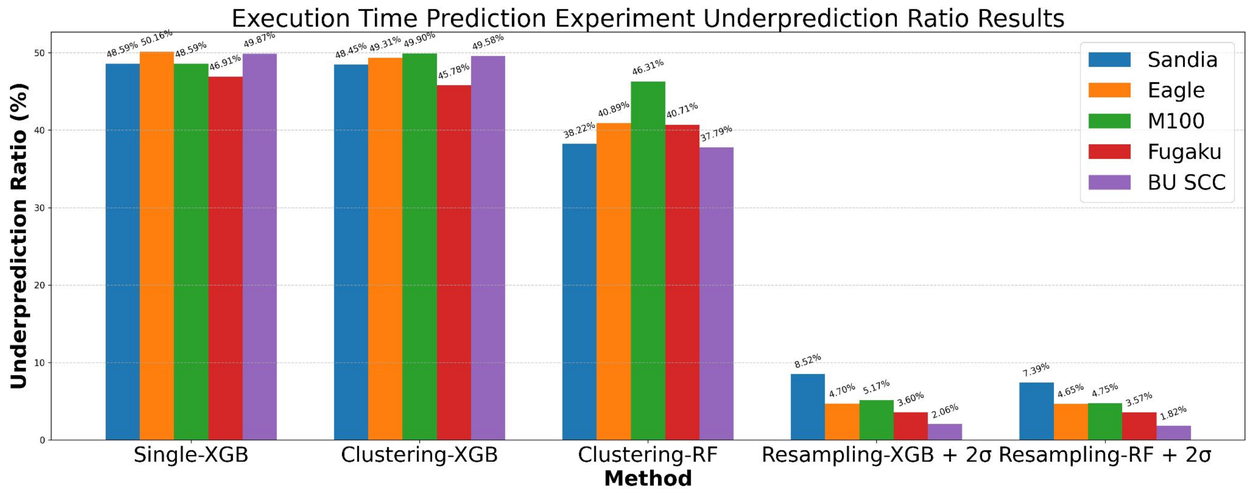
- Training different ML models for each cluster
  - XGBoost & Random Forest Regression
- Time based train-time split configuration
- Performance metrics:

$$\text{Underprediction Ratio} = \frac{\text{Total \# of Underpredictions}}{\text{Total \# of Jobs}}$$

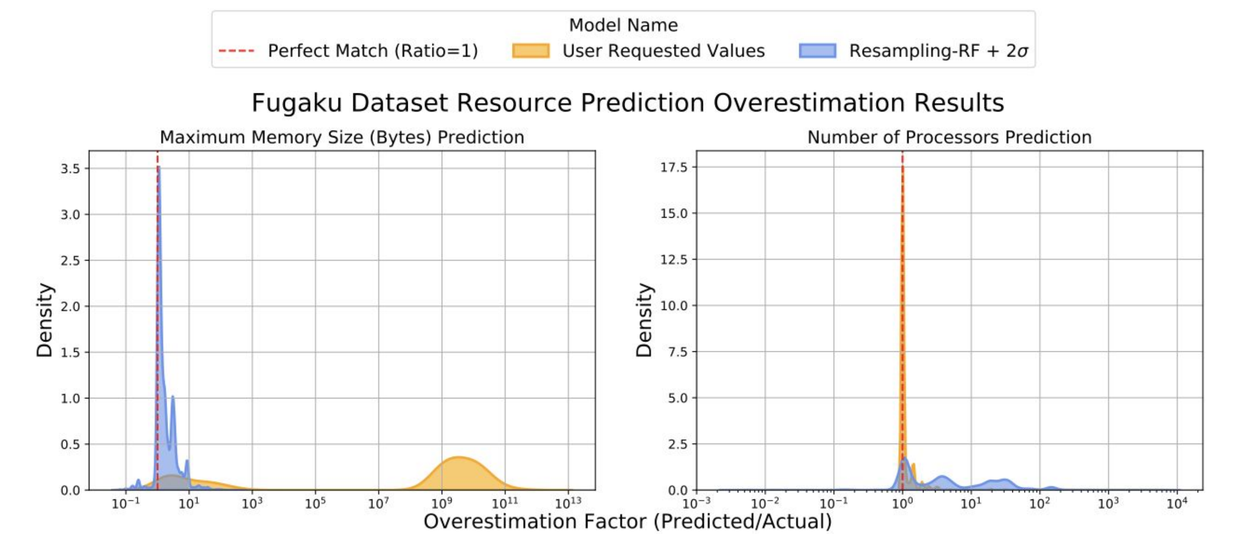
$$\text{Overestimation Factor} = \frac{\text{Resource Prediction}}{\text{Actual Resource Utilization}}$$

Real World Workload Traces
Sandia
NREL Eagle [1]
M100 [2]
Fugaku [3]
BU SCC

Results



The framework predictions show reduction in the underprediction of execution time, memory, and CPU resources to less than 2%.



Overprediction of maximum memory size and number of processor requirements of jobs are reduced compared to the user requested values.

Conclusion & Future Work

- HPC users can benefit from this recommendation system by predicting their jobs' resource requirements before submission.
- We aim to test the framework success on a workload manager simulator and a production HPC system with real user inputs.
- For future work, we will investigate applications' resource utilization variance and the resource contention among compute nodes.

Acknowledgement

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525

References

[1] Menear, K., Nag, A., Perr-Sauer, J., Lunacek, M., Potter, K., & Duplyakin, D. (2023). Mastering HPC runtime prediction: From observing patterns to a methodological approach. Practice and Experience in Advanced Research Computing 2023: Computing for the Common Good (PEARC '23), 75–85. Association for Computing Machinery. https://doi.org/10.1145/3569951.3593598

[2] Borghesi, A., Di Santi, C., Molan, M., et al. (2023). M100 ExaData: A data collection campaign on the CINECA's Marconi100 Tier-0 supercomputer. Scientific Data, 10, 288. https://doi.org/10.1038/s41597-023-02174-3

[3] Antici, F., Bartolini, A., Domke, J., Kiziltan, Z., & Yamamoto, K. (2024). F-DATA: A Fugaku Workload Dataset for Job-centric Predictive Modelling in HPC Systems (1.0) [Data set]. Zenodo. https://doi.org/10.5281/zenodo.11467483