



Introduction

Underpredictions

- Early job terminations
- Reduced system efficiency

Overpredictions

- Increased job waiting time
- Idle resources

Users **manually** request

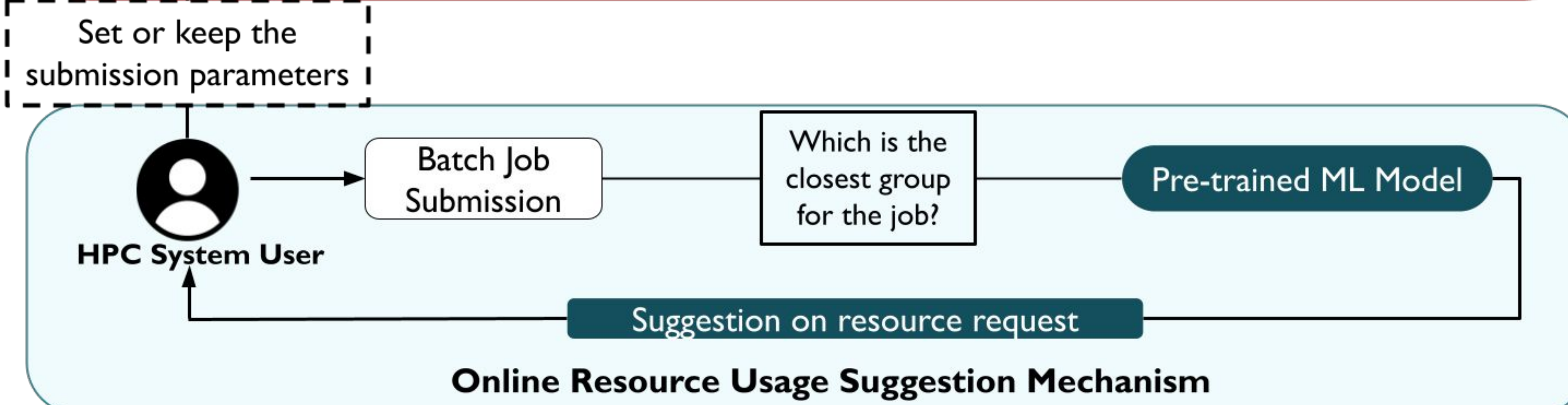
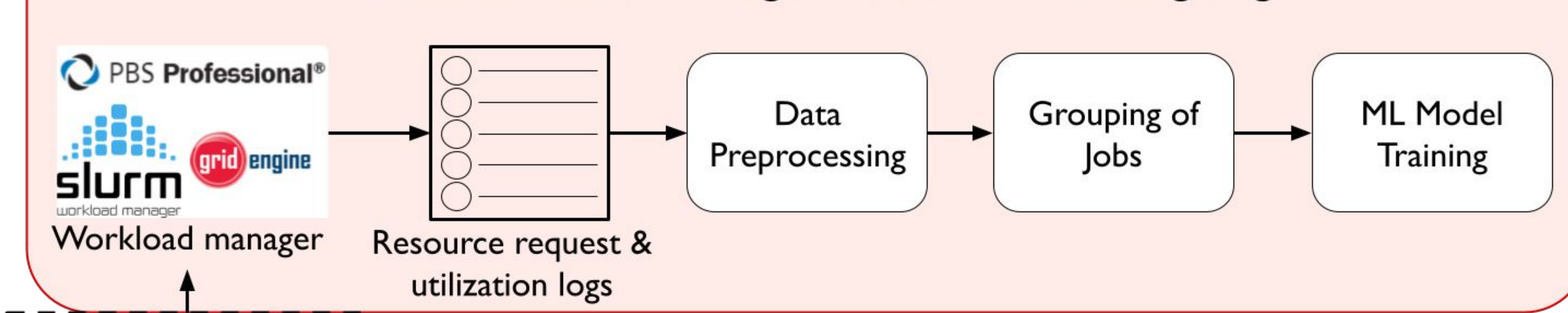
- Execution Time
- Number of Processors
- Maximum Memory Size
- Compute Nodes

Goals:

- Predict HPC batch jobs resources to limit under and overpredictions
- Provide resource recommendations to HPC system users
- Increase overall system utilization and QoS, reduce resource waste

Objectives & Framework Design

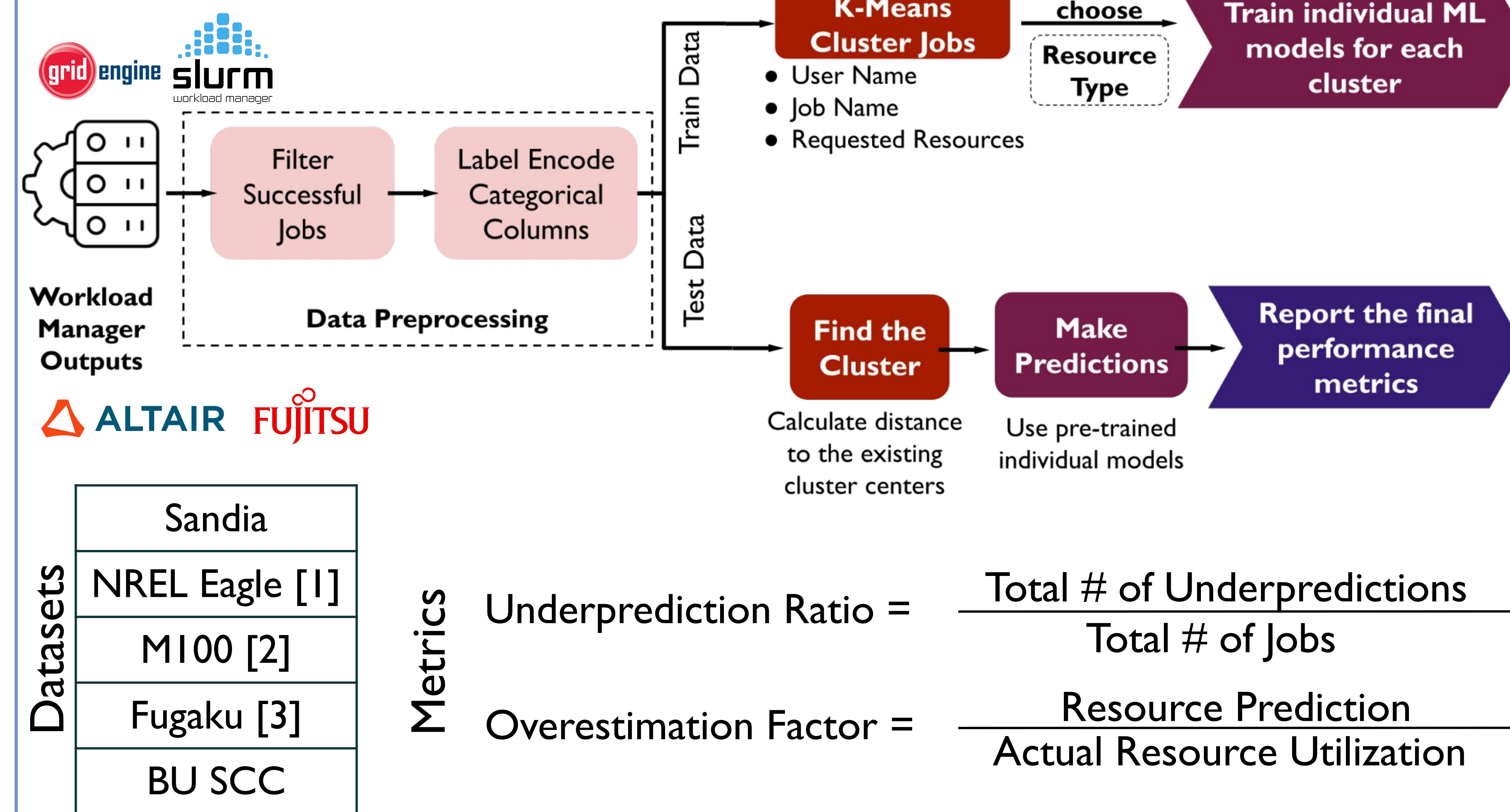
Offline Machine Learning Based Predictor Training Stage



- Model training and testing inputs: batch job submission parameters – such as user, job and project name and requested resources
- Applying data driven strategies after model training to prevent the underestimation of resources

↓ User Wait Time ↑ Job Throughput ↓ CPU Node Hours

Experimental Methodology



Datasets

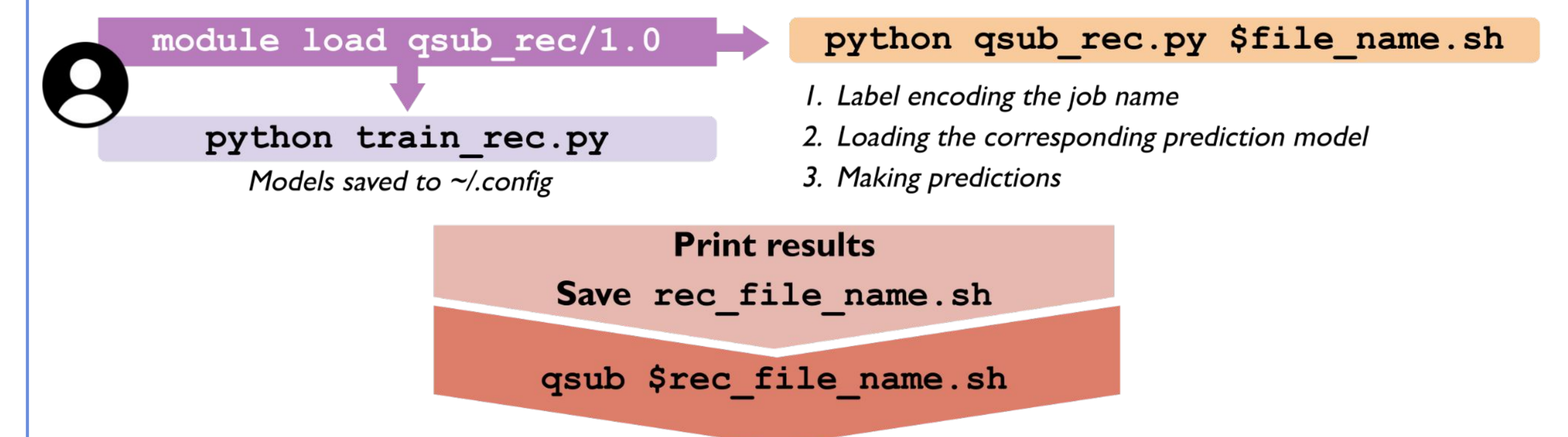
| |
|----------------|
| Sandia |
| NREL Eagle [1] |
| M100 [2] |
| Fugaku [3] |
| BU SCC |

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 System Deployment



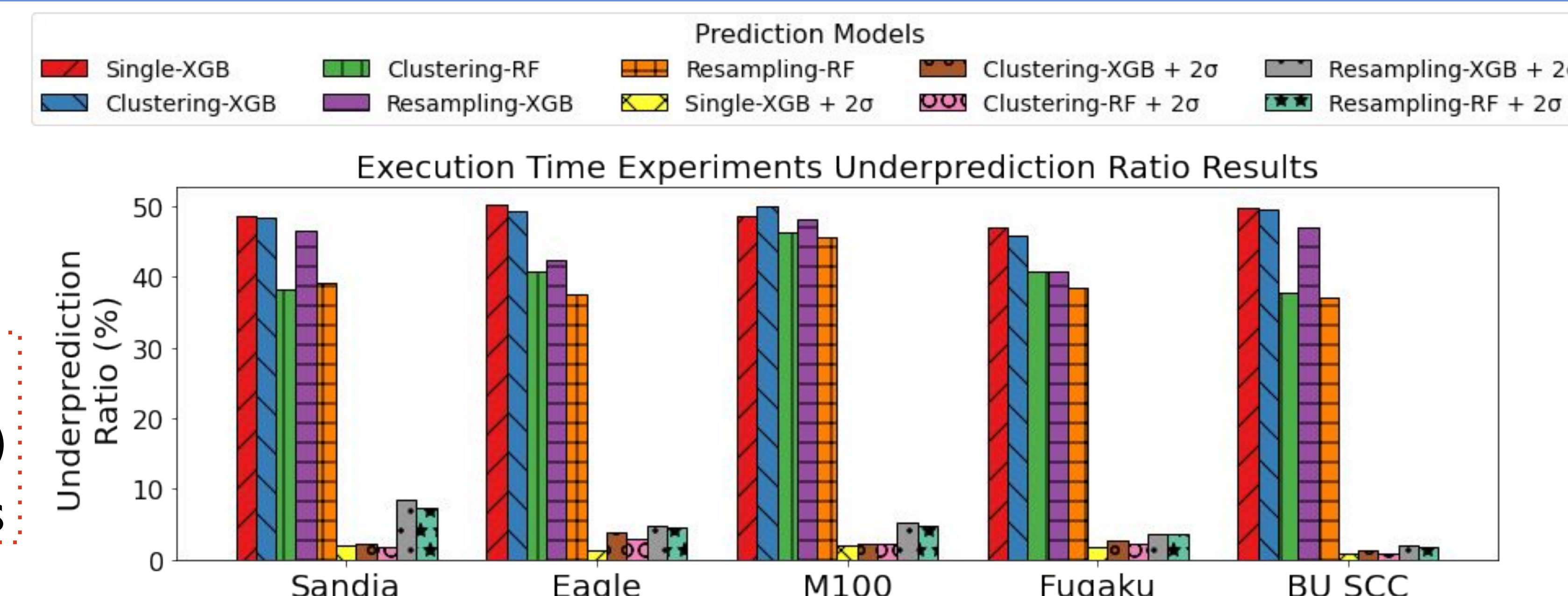
- **Goal:** Provide **online**, data-driven resource request suggestions before jobs are queued
 - An Lmod [4] environment module for Boston University Shared Computing Cluster (BU SCC) users
- Aiming to reduce wait times and resource waste while increasing efficiency and job throughput on BU SCC

Results

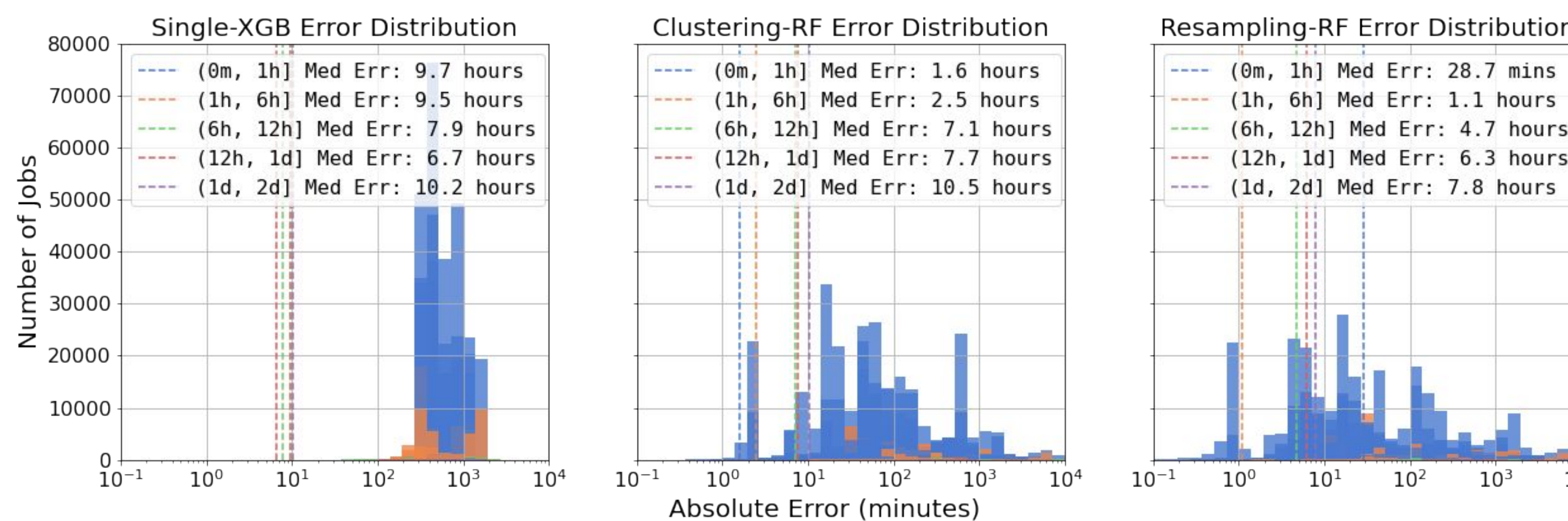
Offline inference experiments

Baseline: Menear et al. [1] XGBoost model

Resampling: Changing the training dataset distribution
Buffering: Adding an offset (2σ) to model predictions



Execution Time Absolute Error Results on Eagle Dataset (with 2σ buffer)



Reducing underprediction ratio of target resource variables to less than 2% and limiting overpredictions.

Conclusion & Future Work

- Job clustering + smaller models results in lower MAE
- Outperforms baseline method
- Towards real-time recommendations to users

- Future work:
 - Test on a production HPC system with real batch jobs
 - Analyze resource usage variance
 - Explore dynamic resource management solutions

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.
- [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.
- [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.
- [4] McLay, Robert, et al. "Best practices for the deployment and management of production HPC clusters." State of the Practice Reports. 2011. 1-11.

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

