Is Knowledge about Running Applications Helping Improve Runtime Prediction of HPC Jobs?

INTRODUCTION

- High-Performance Computing (HPC) systems require **scheduling algorithms**
- Scheduling algorithms **rely on user estimates** of resource requirements
- User estimates are **prone to error**, with runtime usually overestimated
- **Machine Learning models** can be trained to **predict job runtime**
- Knowledge of a job's **primary app** can be established from XALT data
- Only **4 of 20** surveyed papers included primary app in runtime prediction models

Does **incorporating the primary app** improve model performance?

METHODOLOGY

DATA:
3.8 million jobs run on the National Renewable Energy Laboratory (NREL) Eagle HPC system

MODELS:
Two sets of CatBoost models: one set with and one set without including primary application.

TRAIN-TEST SPLIT:
Training/testing sets split at times ranging from Feb. '20 to Sept. '22 in increments of one day, with a 200-day training window and a 24-hour testing window.

TRAINING & PREDICTING:
Two separate models were trained for each split time. Each model was used to predict runtimes for jobs in the corresponding test set.

SHAP VALUES:
SHAP values for each test set were determined and combined to get the average feature importance for all jobs.

(1a) Feature Importance: App Not Included

Feature	Average |SHAP Value|
Wallclock Requested	+11173.41
User	+4446.48
Processors Requested	+3172.15
Partition	+2396.29
Nodes Requested	+922.66
Memory Requested	+883.13
GPUs Requested	+60.37

(1b) Feature Importance: App Included

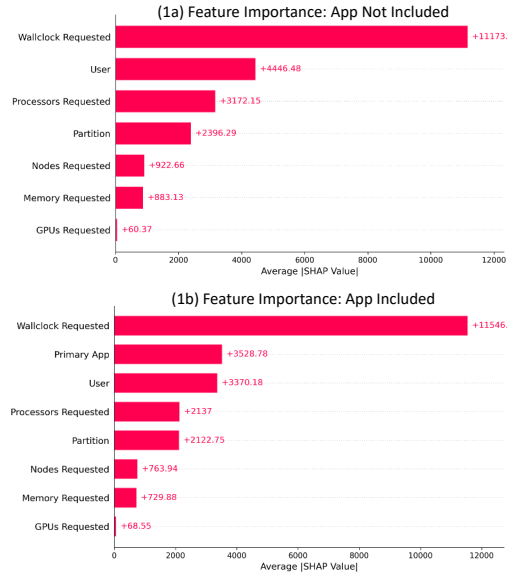| Feature	Average |SHAP Value|
Wallclock Requested	+11546.72
Primary App	+3528.78
User	+3370.18
Processors Requested	+2137
Partition	+2122.75
Nodes Requested	+763.94
Memory Requested	+729.88
GPUs Requested	+68.55

Figure 1: The importance of each feature for the models without the app (top) and with the app (botttom) in terms of their average absolute SHAP value.
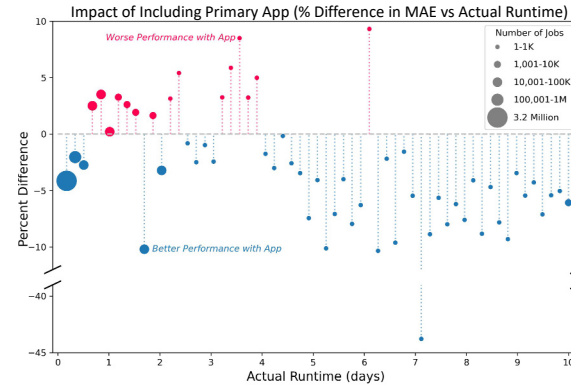

Figure 2: This figure displays the average difference in error between models with jobs binned by the actual runtime (bin width is 4 hours). Most jobs are short, with more than 3.2 million of the 3.8 million jobs contained within the first bin.

Table 1: The models were evaluated with Mean Absolute Error (MAE), Root Mean Squared Error (RMSE) and Coefficient of Variation (R2).

Metric	Without App	With App
MAE (hours)	2.19	**2.12**
RMSE (hours)	7.92	**7.73**
R2	0.444	**0.471**

RESULTS & CONCLUSIONS

- Including the app *improves model performance* (*Table 1*)
- When app is included, it is the second most important feature in the model (*Figure 1b*)
- Performance improvement is most conspicuous for long jobs running more than 4 days (*Figure 2*)
- Improvements are dependent upon the app used by the job (*Figure 3*)
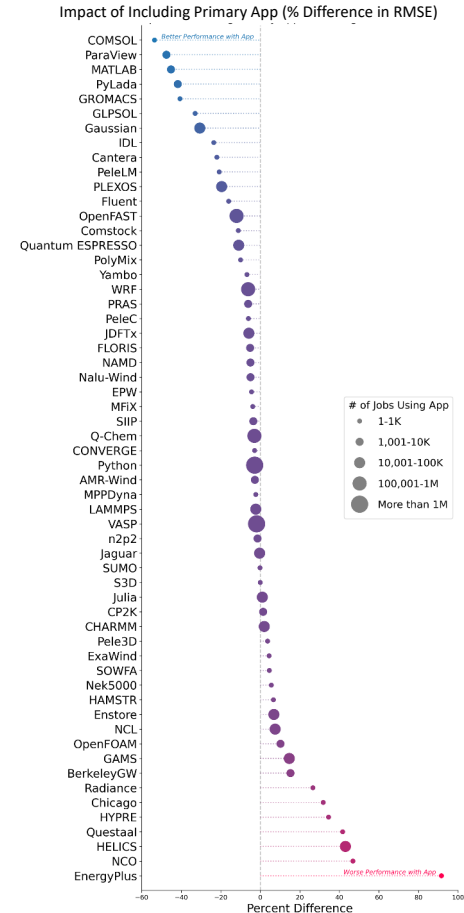

Figure 3: Each marker represents the percentage difference in RMSE when the primary app is included, with results grouped by the app used.

Kevin.Menear@nrel.gov
Dmitry.Duplyakin@nrel.gov