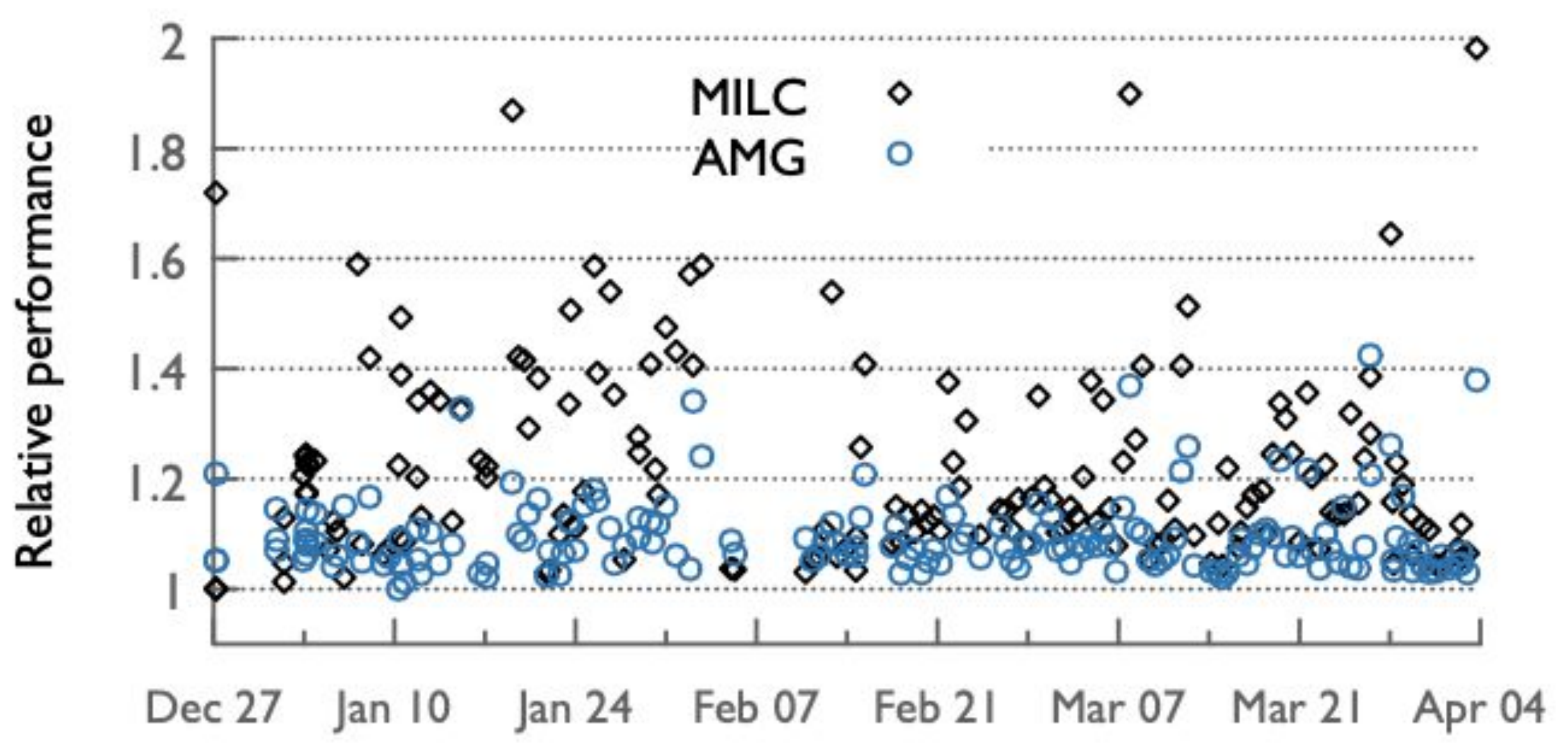
Predicting the Performance of Jobs in the Queue using Machine Learning

Ian Costello, Abhinav Bhatele
University of Maryland, College Park

Motivation



Variability in the performance of 128-node AMG and MILC jobs on different days (on Cori at NERSC).

- Significant job runtime variation isolated to network congestion
- Results in slower science output, lower system throughput, and higher costs.
- This performance variability can be attributed to communication-heavy jobs that contend for network and I/O resources [1], [2].

Data Collection

- Ran ~700 control jobs on Cori @ NERSC over the course of four months. Includes MILC (128 and 512 nodes), AMG (128 and 512 nodes), and miniVite (128 nodes).
- Collected monitoring data via Lightweight Distributed Metric Service (LDMS) for various network counters on each Aries switch across the entire system.
- LDMS data has extremely high dimensionality -- each network switch tracks approximately ~1500 values.

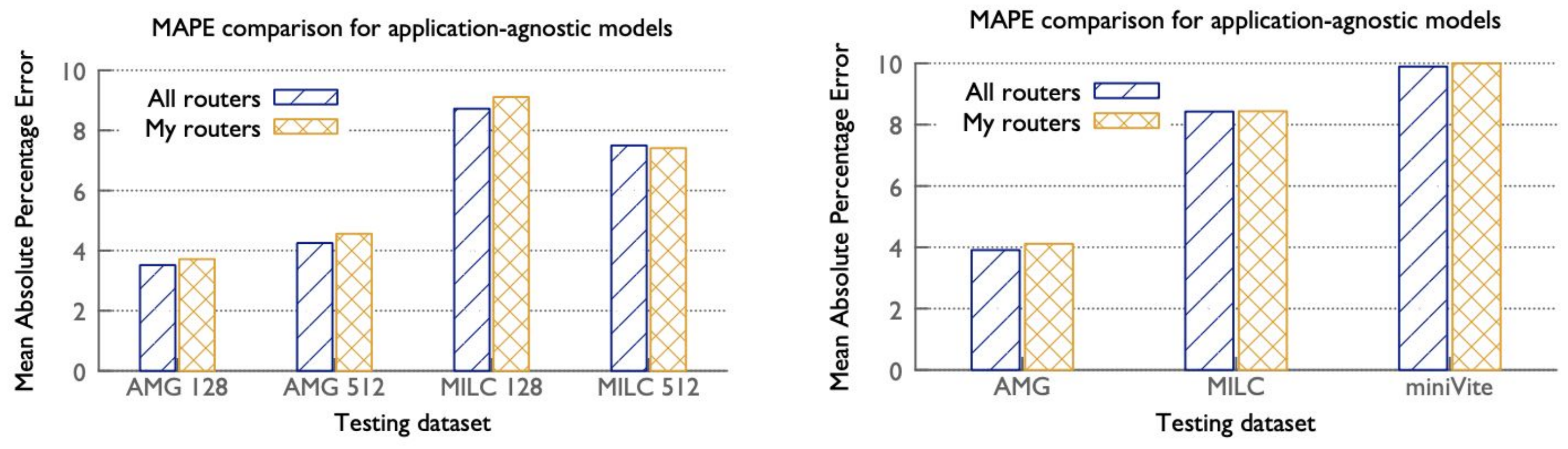
FULL PAPER

PRESENTATION

Inputs to ML Models

- We create a system agnostic pipeline to process, filter and aggregate large-scale system-wide HPC monitoring data to make it suitable for consumption by ML and statistical models.
- The inputs to the ML algorithms for creating the dataset-specific ML models are: (1) For each sample (job) in the training set, values of the aggregated LDMS counters for the five minutes prior to the start of that job are provided as the input features, and (2) Execution time of each sample (job) is provided as the dependent variable to be modeled.

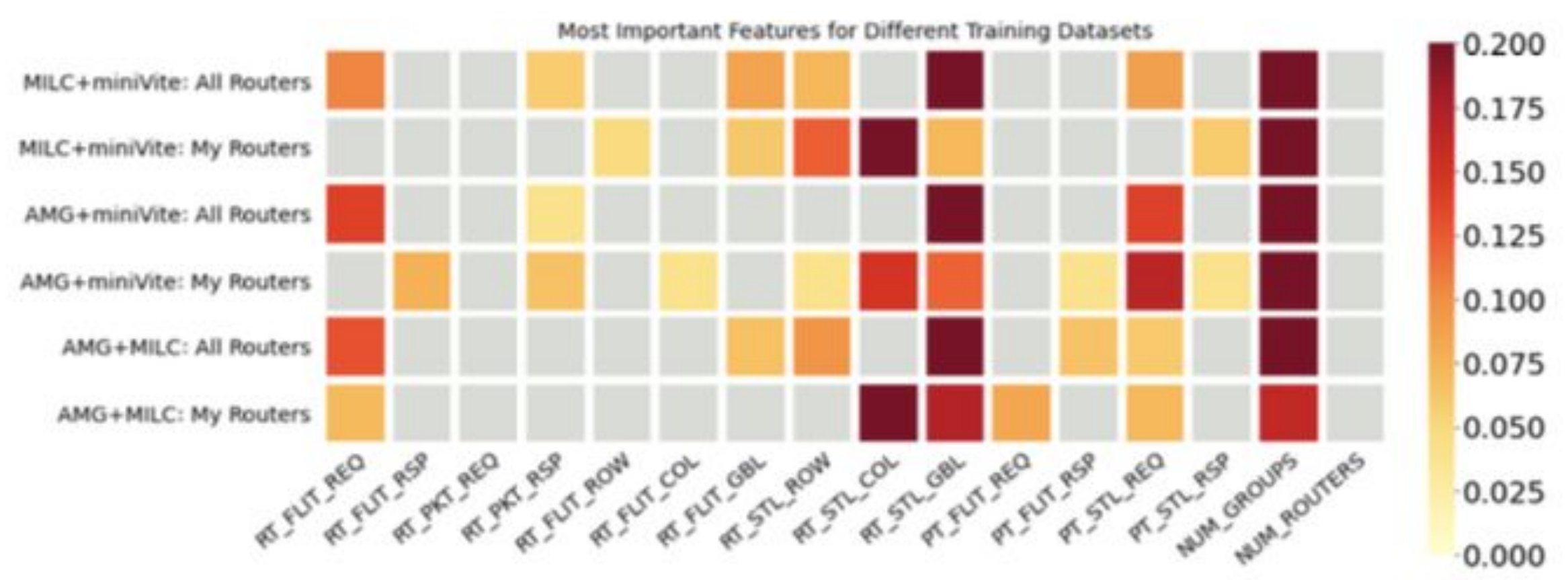
Performance Prediction



MAPE scores for the neural network based model when combining datasets by application type and node counts.

- We develop ML-based regression models that can predict the performance (total execution time) of future jobs using past system state.
- We use both gradient boosting regressors and tuned miniature neural networks and see success with both. Other models were evaluated. Larger and more complex models will likely see more success with more data.
- We show that these models can be application-agnostic and that the benefit of increased data outweighs the performance specificity increase and can strongly predict on unseen applications.
- Strong results given small size of dataset. Seeing promising results as dataset sizes increase.

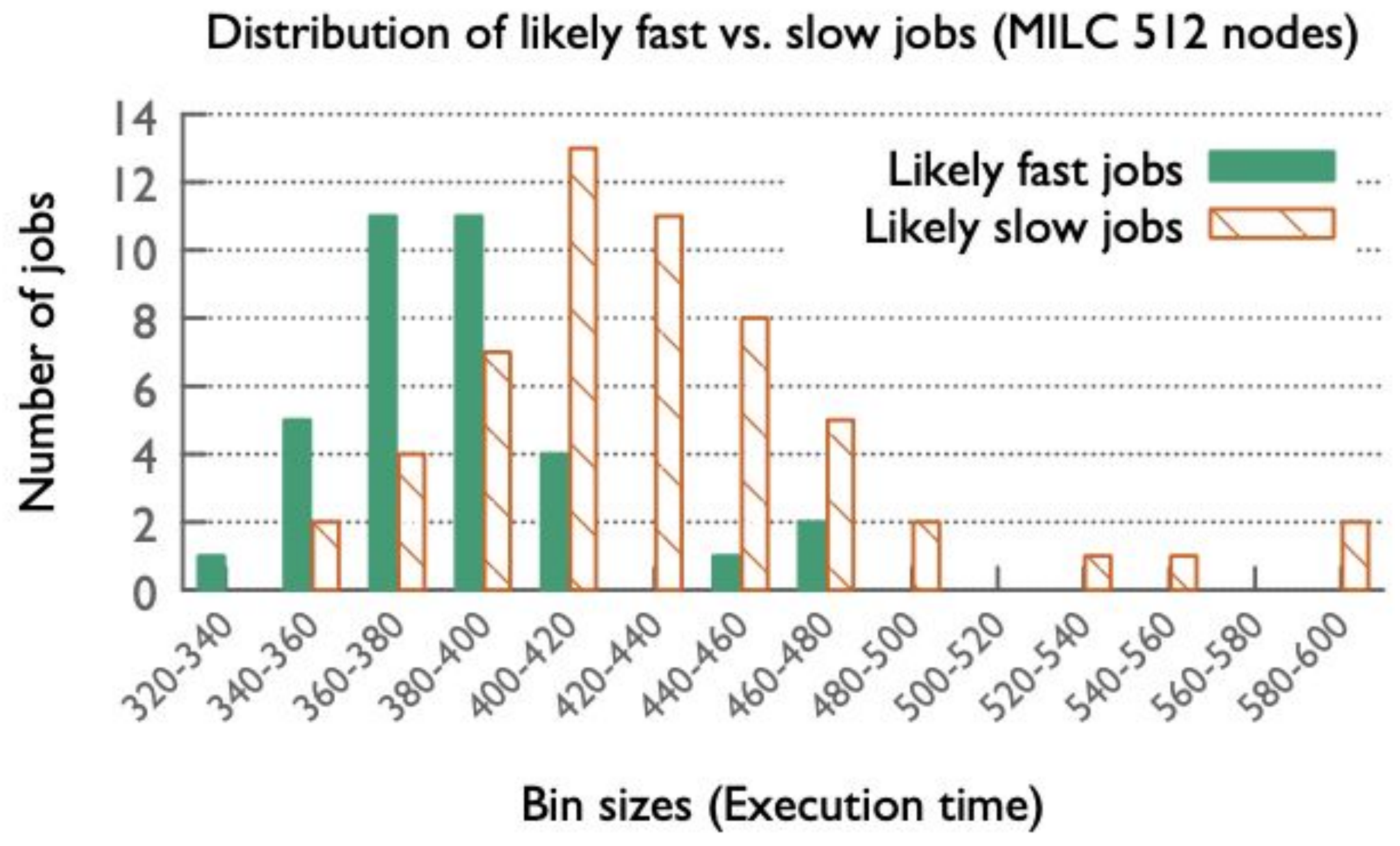
Feature Importances



Relative importances of the most important counters obtained using RFE for different router groups in the application-agnostic model.

- Using recursive feature elimination (RFE) with our regression models, we evaluate the relative importances of network counters in predicting job runtime performance.

Influencing Job Scheduling



Distribution of actual runtimes of likely fast versus slow jobs of MILC when considering above median values of three features: RT STL COL, RT STL GBL, and NUM GROUPS

- We show that we can classify jobs as likely fast or slow based on the values of three counters being all above or below the median value, and the performance of those two groups is statistically different.

Summary

- Created a pipeline to process complex system monitoring data to be digestible by ML.
- Using control jobs, built an application-agnostic performance prediction algorithm for jobs in the queue.

DEPARTMENT OF COMPUTER SCIENCE

References:

[1] A. Bhatele, et al., There goes the neighborhood: performance degradation due to nearby jobs, in *Proceedings of the ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '13. IEEE Computer Society, Nov. 2013.
[2] A. Bhatele, et al., The case of performance variability on dragonfly-based systems," in Proceedings of the IEEE International Parallel & Distributed Processing Symposium, IPDPS '20. IEEE Computer Society, May 2020).