REDWOOD Track I
Workload and Data Management Algorithms
Initial Resource Allocation

Tasnuva Chowdhury (BNL)

DOE ASCR Project : REDWOOD Technical Meeting

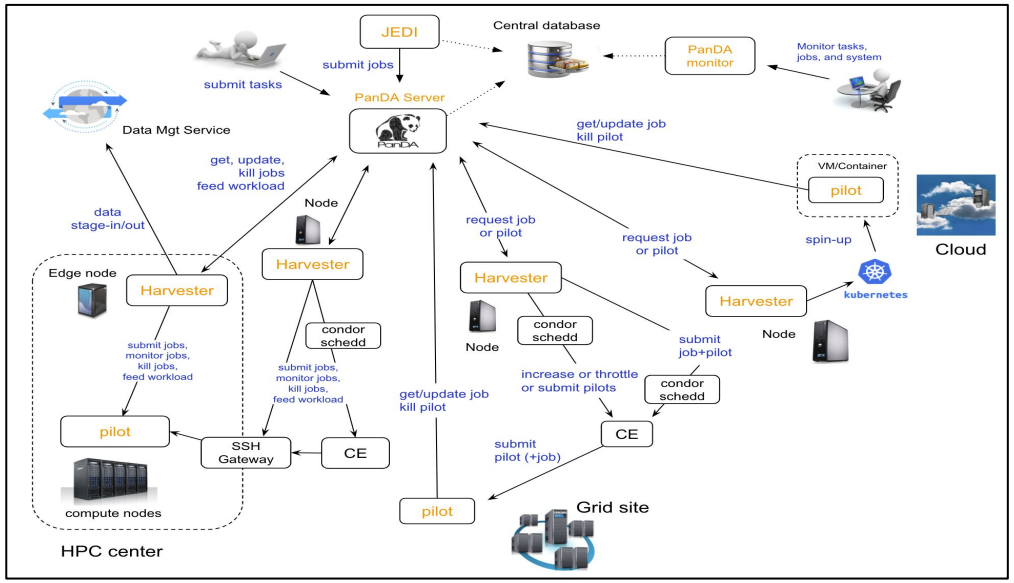
OCT 02, 2024

@BrookhavenLab

tchowdhur@bnl.gov

Track I Objectives

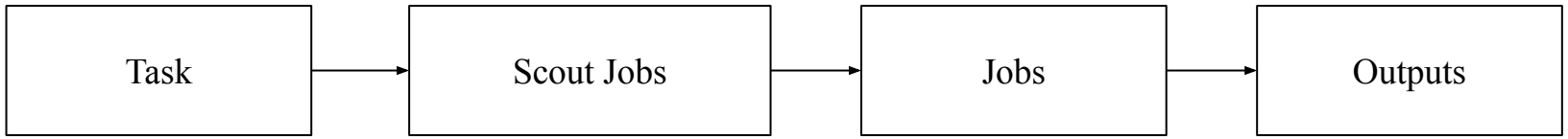
- Current Challenges in PanDA System:
  - Operates in a highly **rule-based**, dynamic environment
  - Manages a **large, complex** set of requirements
  - Results in **time and resource-intensive** operations



- Modernization Strategy:
  - Integrate Machine Learning algorithms into PanDA components
  - Goals: Reduce various processing times, optimize allocation, and enhance system resilience
    - Optimize process for initial resource allocation
    - Input/Output Dataset based on semantics
    - Minimize overall task completion time, while maximizing resource utilization
    - Implement automated error classification and recovery procedures

Fundamental Units of PanDA Workflow

- A task is a unit of workload for a physics objective, involving data input/output
- A job is a sub-unit of a task, designed to execute on allocated computing resources based on preferences, resource constraints and job metrics. Jobs process subsets of task input to produce outputs collectively forming the task output
- Scout jobs are a subsample of jobs which calculate resource requirements based on job metrics (e.g memory, cpu-time, cpu efficiency, io-intensity). Their completion initiates the execution of remaining jobs with more accurate resource requirements



Task & Job Status

Task Status Flow: Involves following stages

- Initiation (registered, defined, assigning resources)
- Preparation (ready, **scouting, scouted**)
- Execution (running, prepared)
- Completion (done, failed/finished)
- Interruption (aborting/aborted)
- Additional Processes (preprocess, toretry)

Job Status Flow:

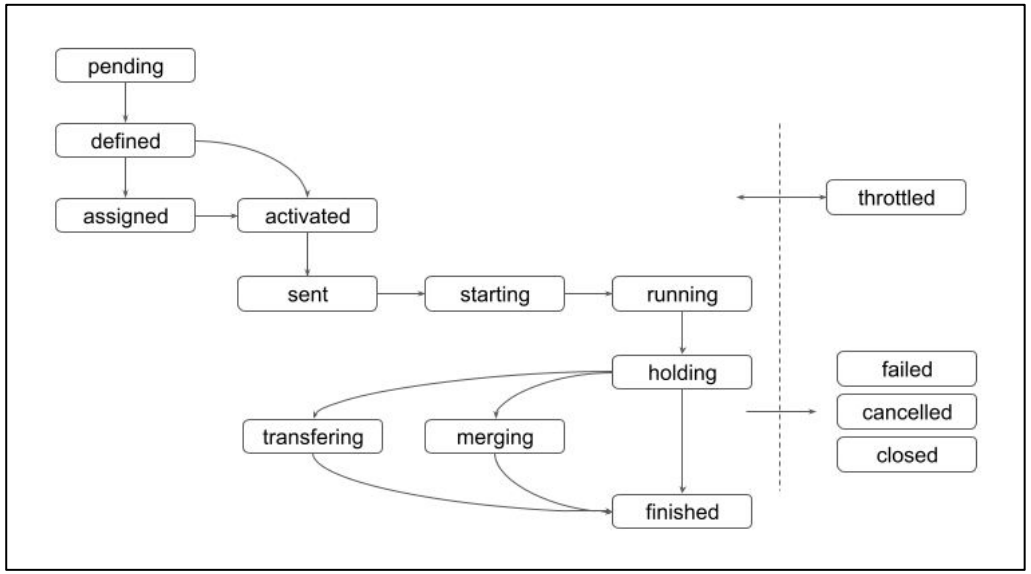
- Initialization (pending, defined, assigned)
- Activation (activated, sent, starting)
- Execution (running, holding, merging, transferring)
- Final States (finished, failed, closed, cancelled)

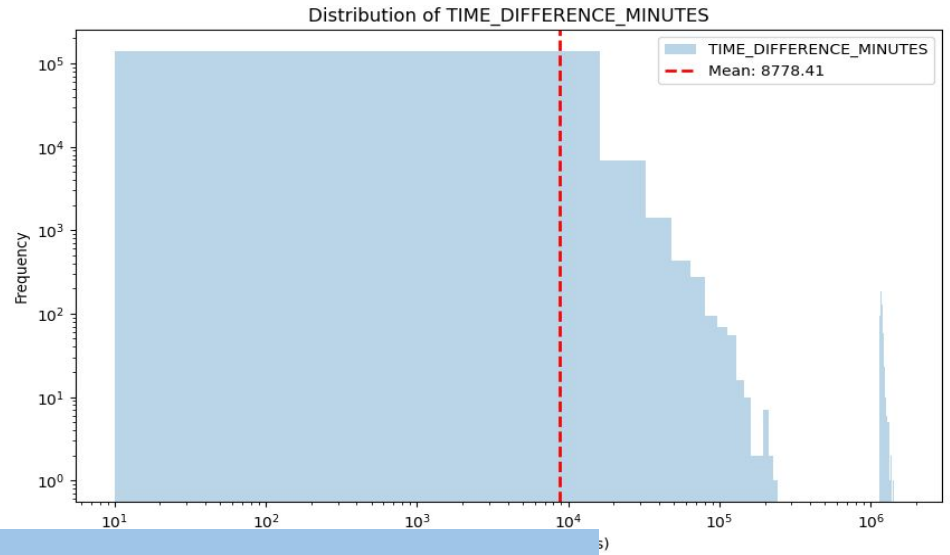
Scout Jobs

Scout Jobs are early jobs assigned to computing resources based on preliminary resource requirements, following the same status flow as regular jobs

Caveats:

- Jobs can wait for long time in the queue to be initiated
- The transition of a Task from scouting to scouted can take several hours due to queuing time and execution time of scout jobs, and retries necessitated by occasional failures
- If all scout jobs (from a task) fail, resource requirements do not get updated, resulting in early stage failure





Time difference is calculated between the task status **scouting** & **scouted** on 150,000 tasks

Brookhaven National Laboratory

Target Variables From Scout Jobs

Current brokerage relies on these metrics to split task into jobs

$$cpuTime = \frac{max(0, endTime - startTime - baseTime) \times corePower \times coreCount \times cpuEfficiency \times 1.5}{nEvents}$$

The estimated walltime for a job is

$$\frac{cpuTime \times nEvents}{C \times P \times cpuEfficiency} + baseTime$$

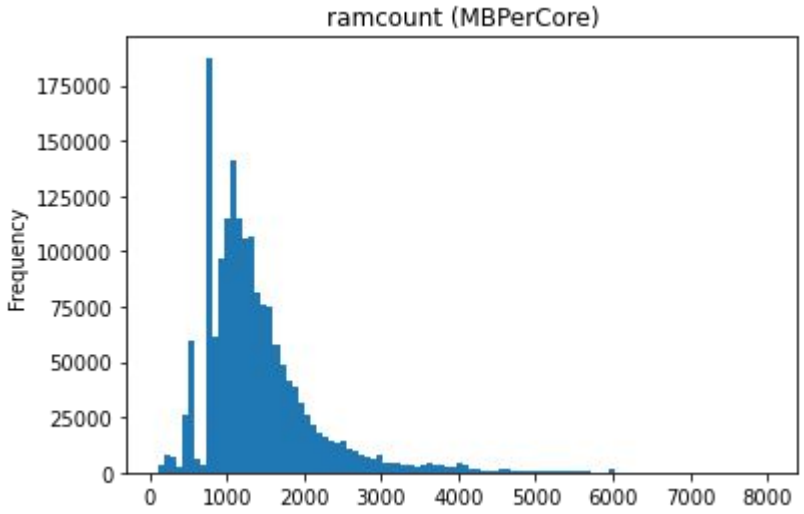
$$S \geq inputDiskCount + max(0.5GB, outDiskCount \times nEvents) + workDiskCount$$

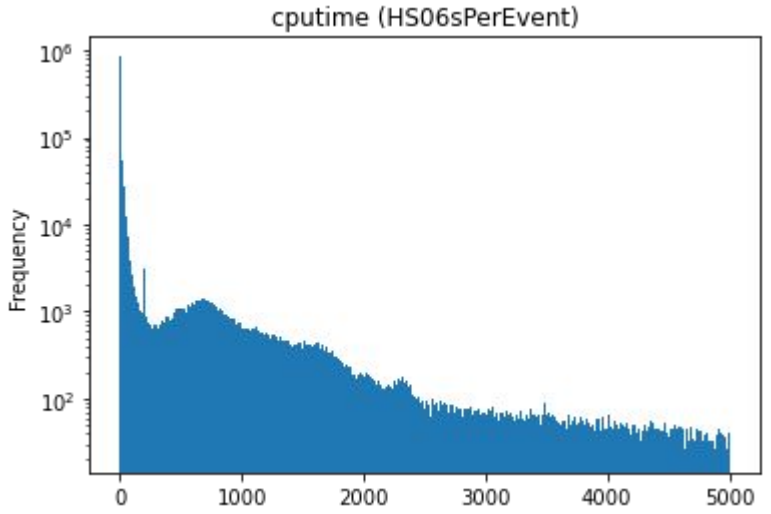
$$W \geq \frac{cpuTime \times nEvents}{C \times P \times cpuEfficiency} + baseTime$$

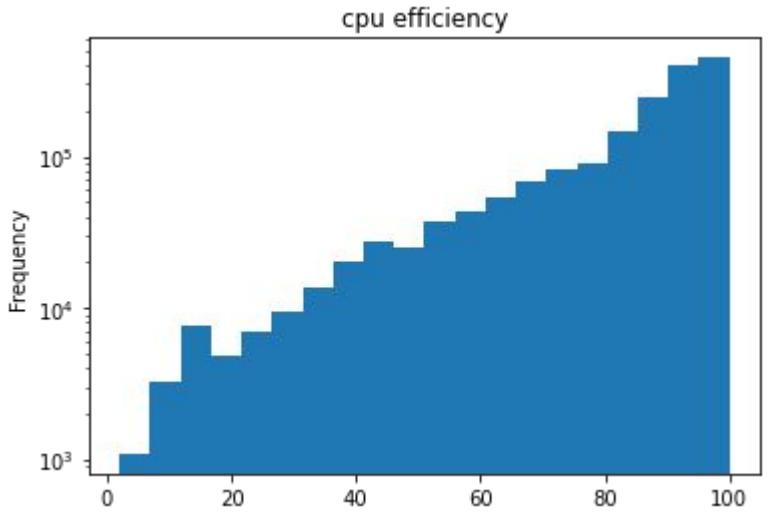
where *S*, *W*, *C*, and *P* are the scratch disk size, the wall time limit, the number of CPU cores, and the HS06 core-power at the computing resource, respectively. *inputDiskCount* is the total size of job input files, a discrete function of *nEvents*. Note that *inputDiskCount* is zero if the computing resource is configured to read input files directly from the local storage resource.

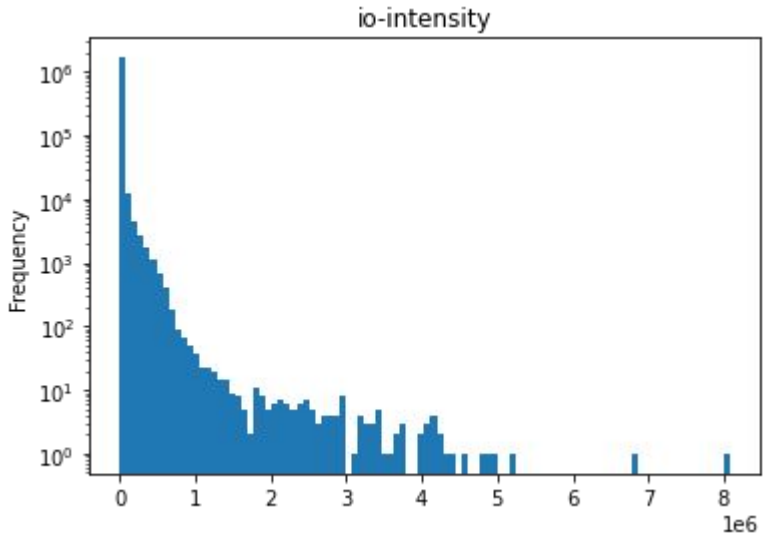
Model	Target
1	ramcount
2	cpu time
3	cpu efficiency
4	iointensity

Target Variables From Scout Jobs & Features









Categorical Features:

- TaskType
- Core
- Processing
- Framework

Numerical Features:

- nEvents
- nFiles
- DatasetName(count)

Data Sample:

- 3 years data ~ 2.2M Tasks
- After removing outliers ~1.1M Tasks

Brookhaven National Laboratory

Feature Description

Categorical Features:

- TaskType (production or analysis task)
- Core (single or multi core)
- Processing (physics process: derivation, reconstruction, simulation, etc.)
- Framework (different code packages for physics processes Athena, Athsimulation, etc)

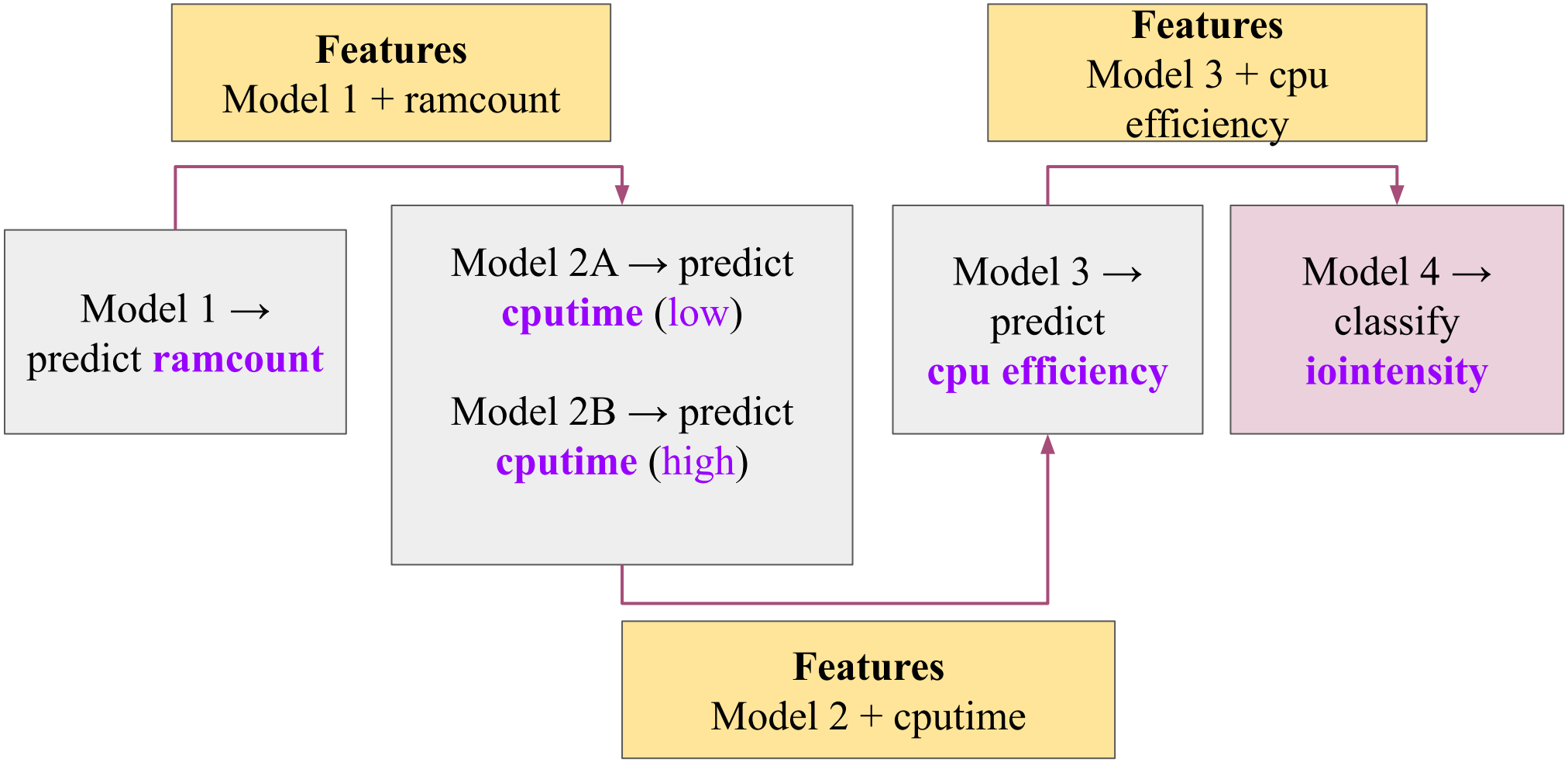
Numerical Features:

- nEvents (Total no of events in the input dataset/datasets for a task)
- nFiles (Total no of Files)
- DatasetName(How many unique input dataset)

Data Sample:

- 3 years data ~ 2.2M Tasks
- After removing outliers ~1.1M Tasks

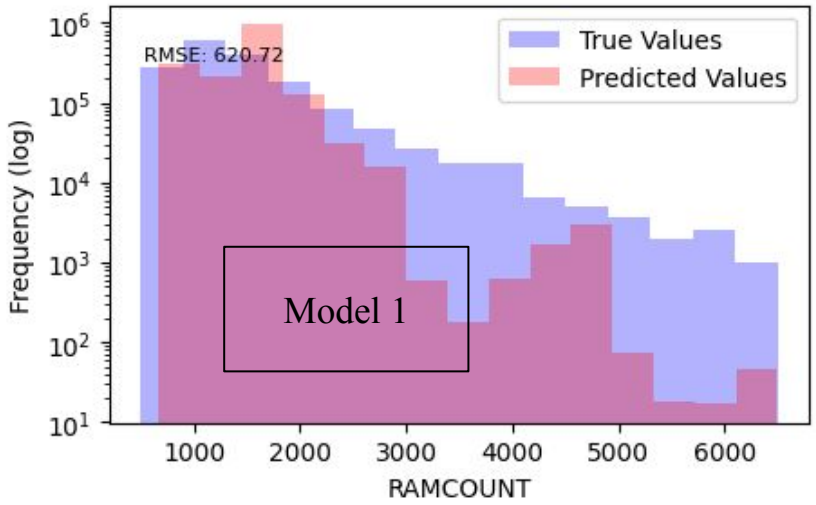
Sequence of Models

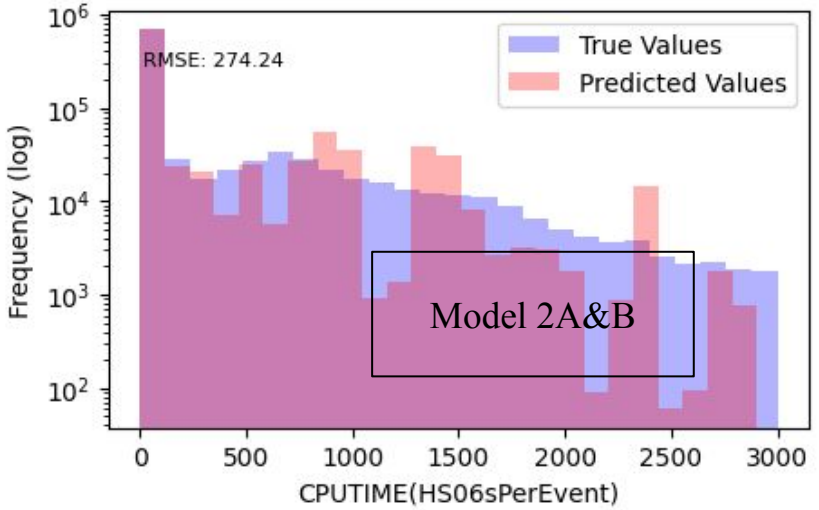


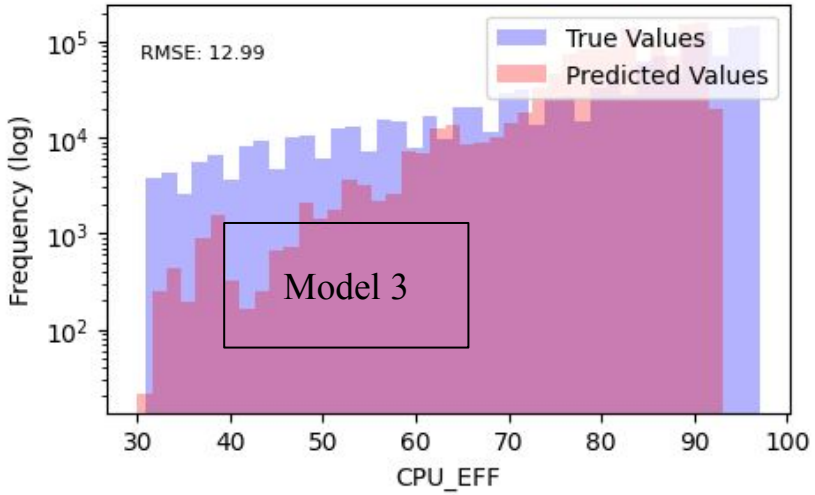
Training and Validation is done with real feature & target values

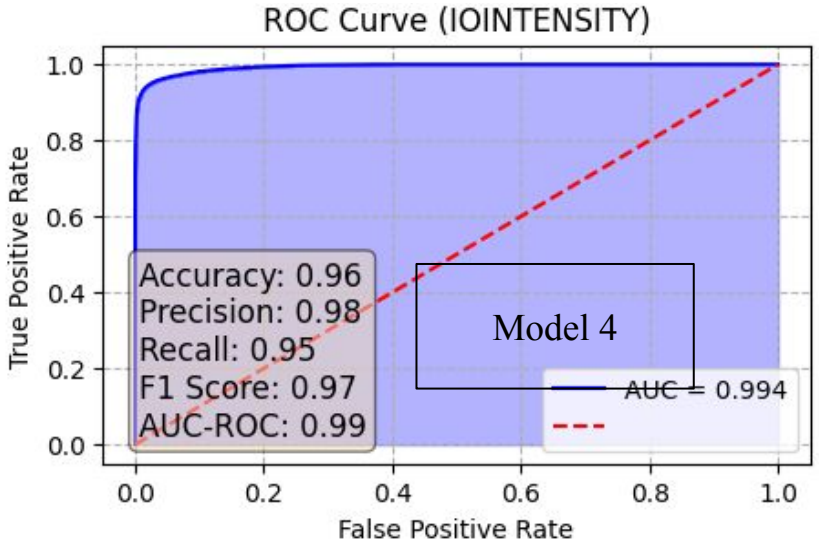
Test is done using predicted target values from previous models in the sequence

Overall Model Performance









- 5 deep learning models to predict →

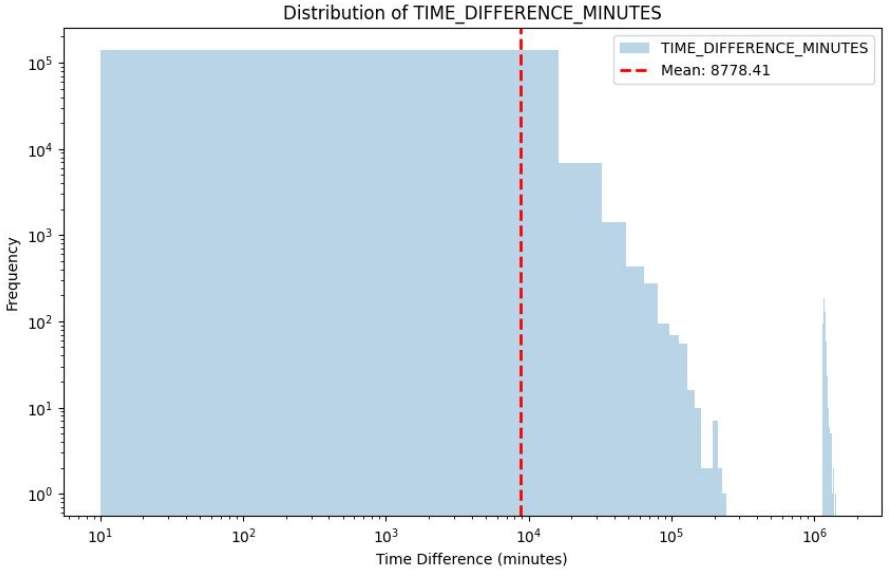
  4 target variables

Summary

- ✓ *5 deep learning models to predict 4 target variables*
- ✓ *Prediction outperforms scout jobs success rate*
- ✓ *Helps to avoid using default values when scouts fail*
- ✓ *Save hours per task by avoiding scouting phase*
- ✓ *The prediction will accurately determine resource requirements for each workload upfront, an crucial building block in dynamically optimizing resource usage and enhancing system resilience*

Work In Progress

- Deployment in testing environment
- Automated model pipeline
- MLFLOW for tracking ML experiment and model registry (first time implementation in ATLAS)



Future Goals

- Real time prediction
- Deploy in production with error monitoring
- Explore Data semantics
- Automated error classification and recovery procedures

Brookhaven National Laboratory

Thank You