ML-based Computer System Telemetry Analytics

BY BURAK AKSAR[1], MARTIN MOLAN[2], ANDREA BORGHESI[2], LUCA BENINI[2], ANDREA BARTOLINI[2], AYSE K. COSKUN[1]

[1]Boston University; [2]University of Bologna

August 30, 2022

AGENDA

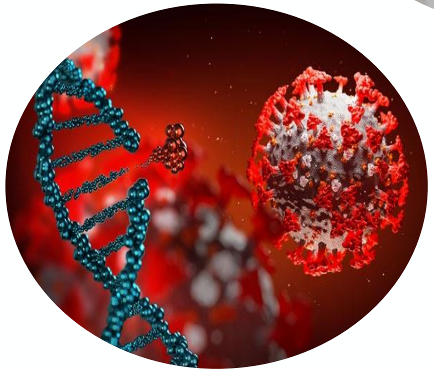
- **Overview: Telemetry Data-based Analytics on Large-scale Computing Systems**
- Supervised Methods: Anomaly Detection and Diagnosis
- Break
- Semi-supervised and Unsupervised Anomaly Detection in Supercomputers
- Deployment: Challenges and Current Status
- Hands-on Activity

HIGH PERFORMANCE COMPUTING (HPC) SYSTEMS

Fugaku

Summit

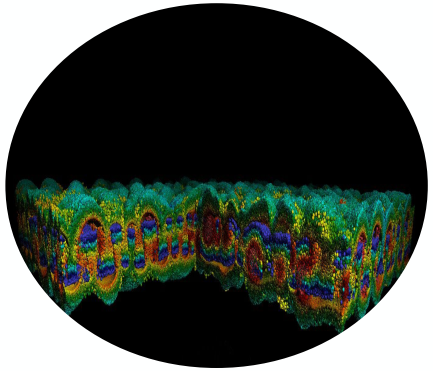
Sierra



Medicine

Security

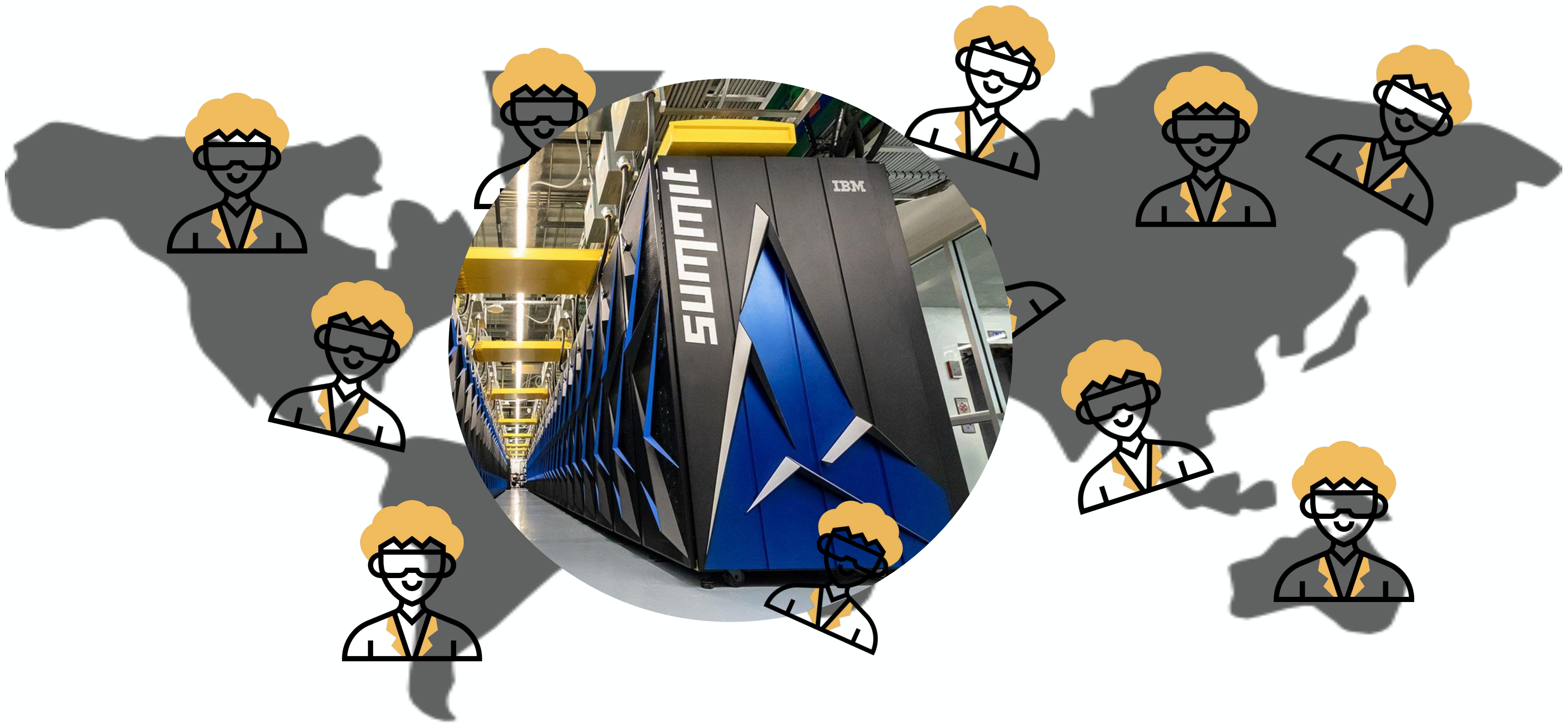
Climate



Simulations

https://innovationatwork.ieee.org/cyber-security-advancing-through-ai/
https://www.sandia.gov/news/publications/labnews/articles/2020/05-08/COVID-19_CRISPR.html
https://www.japantimes.co.jp/news/2020/06/23/national/fugaku-supercomputer-ranked-fastest/
https://en.wikipedia.org/wiki/Sierra_(supercomputer)
https://phys.org/news/2018-06-ornl-summit-supercomputer.html
https://cars.okstate.edu/

HIGH DEGREES OF RESOURCE SHARING



PERFORMANCE VARIATIONS IN HPC

Up to **8x** delay

in job execution time [Zhang et al.,Cluster'20]

70% variation

in application performance [Chunduri et al., SC'17]

Performance variations in HPC

- Variable running time
- Job terminations
- Wasted computing power
- Performance degradation

PERFORMANCE VARIATIONS IN HPC

How do we detect and diagnose performance anomalies?

TELEMETRY DATA-BASED PERFORMANCE ANOMALY DIAGNOSIS

- Terabytes of telemetry data per day
  - Logs, performance metrics, traces, etc.
- Rule-based anomaly detection methods are commonly deployed in large-scale systems
  - Threshold-based rules on the monitored resource usage, metrics, and applications [Ahad et al., ICCAC'15; Jayathilaka et al., WWW'17]
- Disadvantages:
  - Focused on **detecting** anomalies - no information about the root cause
  - Reliance on expert knowledge
  - Dependence on the target HPC infrastructure

TELEMETRY DATA-BASED PERFORMANCE ANOMALY DIAGNOSIS

- Machine learning (ML) frameworks
  - Raise an anomaly alert whenever the prediction does not match the forecasted metric value beyond an acceptable range of differences
    - Support vector regression [Jin et al., ITC'16]
    - Autoregressive moving averages [Laptev et al., SIGKDD'15]
    - Holt-Winters forecasting [Nair et al., SIGKDD'15]

ML framework: HPC performance analytics tools that leverage ML approaches

TELEMETRY DATA-BASED PERFORMANCE ANOMALY DIAGNOSIS

- Machine learning (ML) frameworks
  - Detect & diagnose performance anomalies and suspicious behaviors
    - Density estimation [Baseman et al., SIGKDD'16]
    - Random Forest [Klinkenberg et al., Cluster'17]
    - Autoencoder [Borghesi et al., IAAI'20]
    - Autoencoder + SVM [Aksar et al., ISC'21]

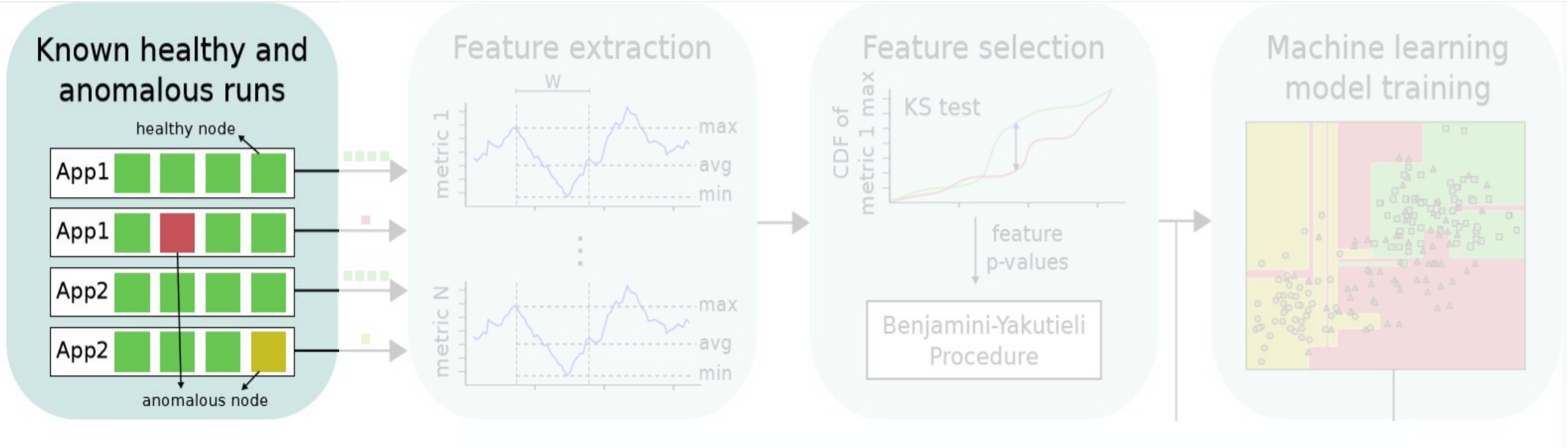
ML framework: HPC performance analytics tools that leverage ML approaches

AGENDA

- Overview: Telemetry Data-based Analytics on Large-scale Computing Systems
- **Supervised Methods: Anomaly Detection and Diagnosis**
- Break
- Semi-supervised and Unsupervised Anomaly Detection in Supercomputers
- Deployment: Challenges and Current Status
- Hands-on Activity

ONLINE DIAGNOSIS OF PERFORMANCE VARIATION IN HPC SYSTEMS USING MACHINE LEARNING

Tuncer et al., ISC'17, Gauss Award
Tuncer et al., TPDS'18



- LDMS *[Agelastos et al., SC'12]* data is collected from the applications once per second
  - Hardware counters, memory/CPU usage metrics, etc.
  - 100s of time series per node

HPC SYSTEMS

ECLIPSE

Benchmark	Application	Description
Real Applications	LAMMPS	Molecular dynamics
	HACC	Cosmological simulation
	sw4	Seismic modeling
ECP Proxy Suite	ExaMiniMD	Molecular dynamics
	SWFFT	3D Fast Fourier Transform
	sw4lite	Numerical kernel optimizations

- Production HPC system
- 1488 compute nodes
- Run applications:
  - 4 nodes
  - 20 – 45 mins

VOLTA

Benchmark	Application	Description
NAS	BT	Block tri-diagonal solver
	CG	Conjugate gradient
	FT	3D Fast Fourier Transform
	LU	Gauss-Seidel solver
	MG	Multi-grid on meshes
	SP	Scalar penta-diagonal solver
Mantevo	miniMD	Molecular dynamics
	CoMD	Molecular dynamics
	miniGhost	Partial differential equations
	miniAMR	Stencil calculation
Other	Kripke	Particle transport

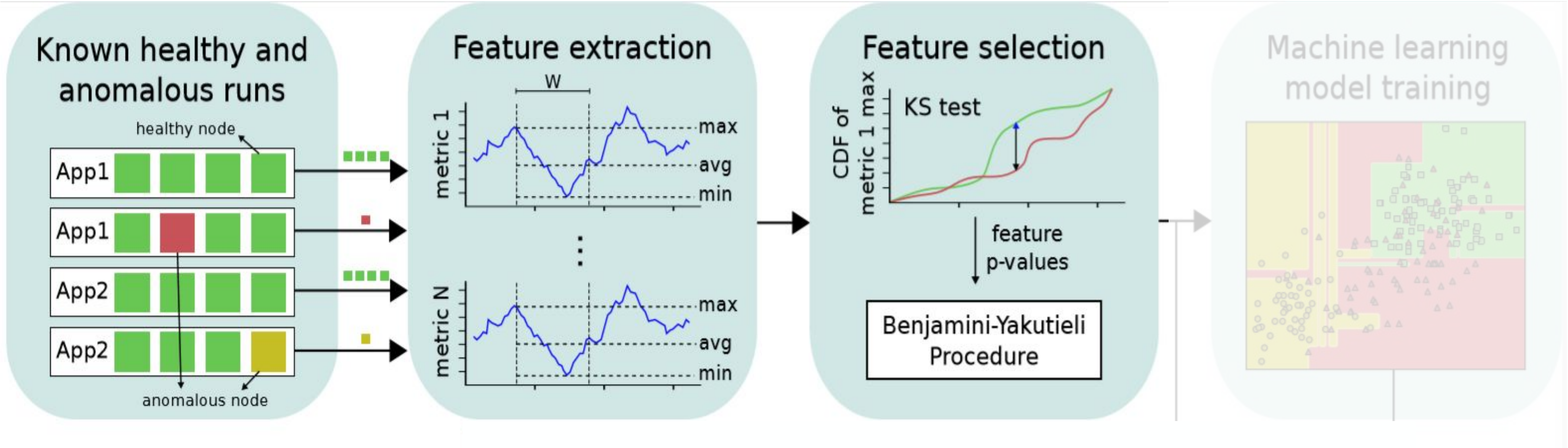
- Testbed HPC system
- 52 compute nodes
- Run applications:
  - 4, 8, and16 nodes
  - 10 – 15 mins

SYNTHETIC ANOMALIES

Anomaly type	Anomaly behavior	Configuration
CPU intensive process	Arithmetic operations	-u 100%, 80%
Cache contention	Cache read & write	-c L1, L2 / -m 1, 2
Memory bandwidth contention	Uncached memory write	-s 4K, 8K, 32K
Memory leakage	Increasingly allocate & fill memory	-s 1,3,10 M / -p 0.2,0.4,1

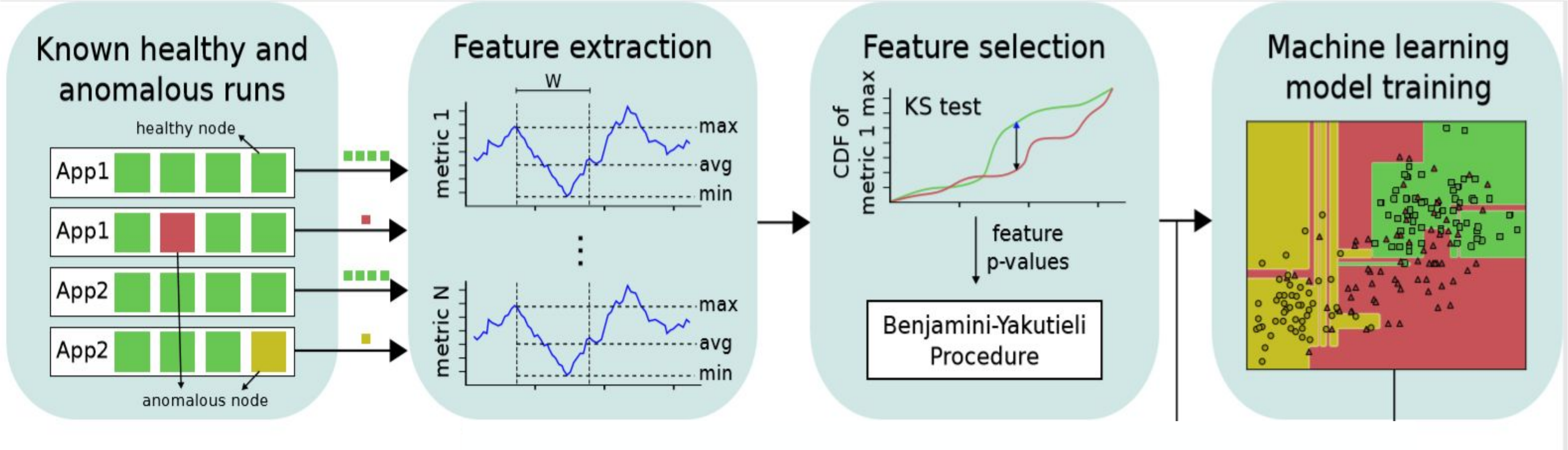
- High-performance anomaly suite (HPAS) mimics performance anomalies observed in large-scale systems *[Ates et al., ICPP'19]*
  - Anomalies have multiple intensities and target a specific subsystem

PERFORMANCE VARIATION DIAGNOSIS - METHODOLOGY



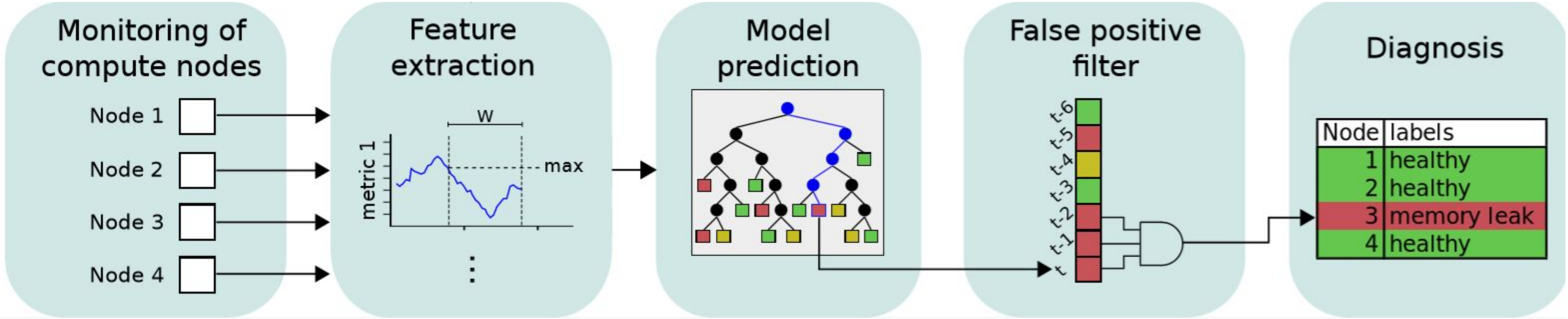
- To reduce data dimensionality, statistical properties of the timeseries are extracted
- Feature selection to choose important features and achieve lower overhead for runtime analysis

PERFORMANCE VARIATION DIAGNOSIS - METHODOLOGY



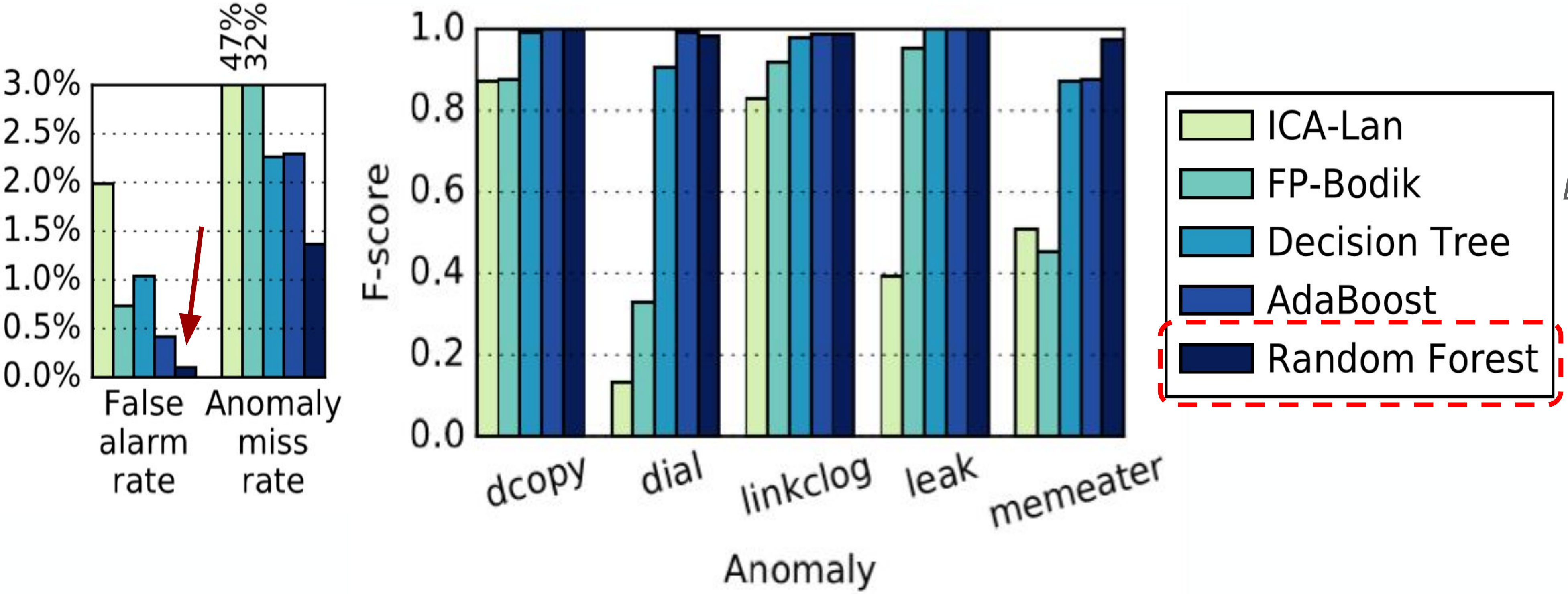
- Models: Random Forest, Decision Tree, SVM, Extra Trees

PERFORMANCE VARIATION DIAGNOSIS - METHODOLOGY



- At runtime:
  - Features are extracted with 45-seconds window
  - False positive filter prevents glitches in the consecutive time windows
  - Use the models trained in the offline phase

PERFORMANCE VARIATION DIAGNOSIS - EVALUATION



[Lan et al., TPDS'10]

[Bodik et al., EuroSys'10]

- Diagnosed **98%** of the injected performance anomalies with **0.3%** false alarm rate

LIMITATIONS OF THE EXISTING ML FRAMEWORKS

- Two major problems:
  - Collecting labeled telemetry data is challenging
    - Unknown applications and application inputs
    - Thousands of compute nodes
  - Most supervised frameworks require a large set of labeled telemetry data
    - After collecting telemetry data, most of the labeling solutions are heuristic-based

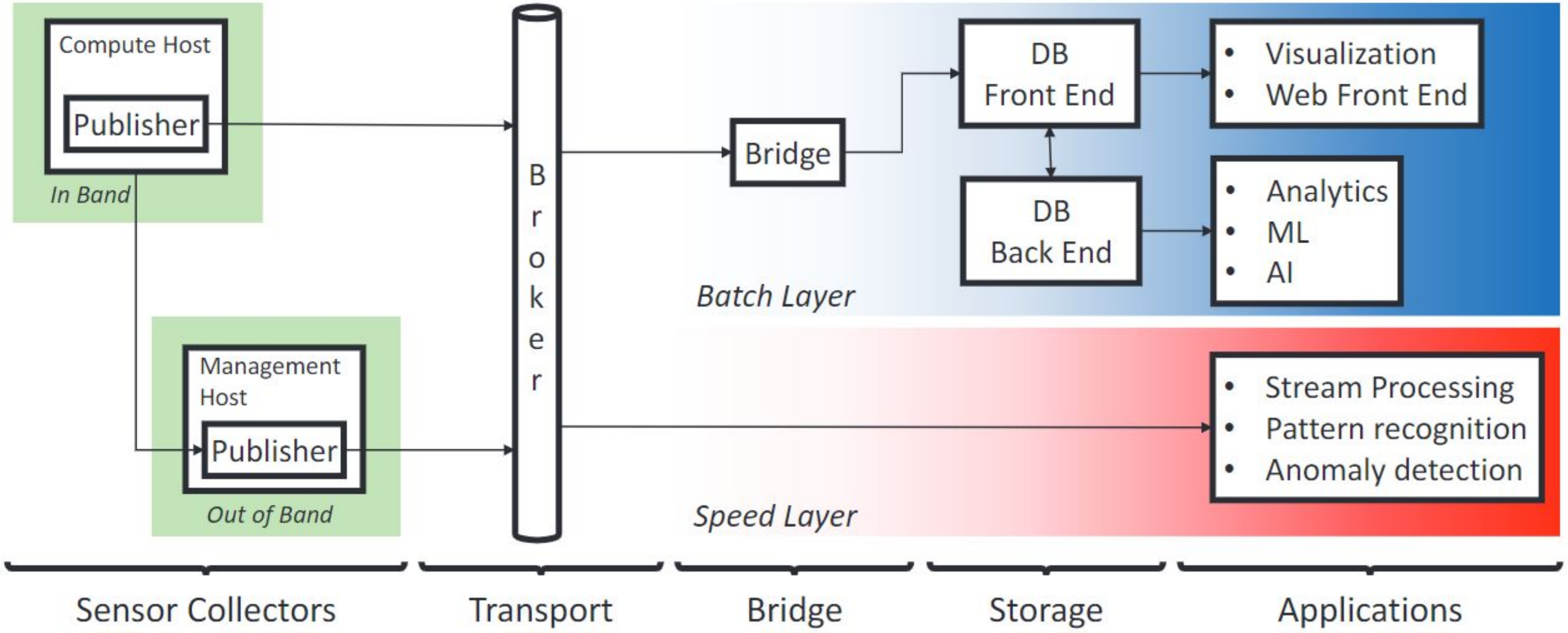
AGENDA

- Overview: Telemetry Data-based Analytics on Large-scale Computing Systems
- Supervised Methods: Anomaly Detection and Diagnosis
- Break
- **Semi-supervised and Unsupervised Anomaly Detection in Supercomputers**
- Deployment: Challenges and Current Status
- Hands-on Activity

MONITORING AND DATA ACQUISITION

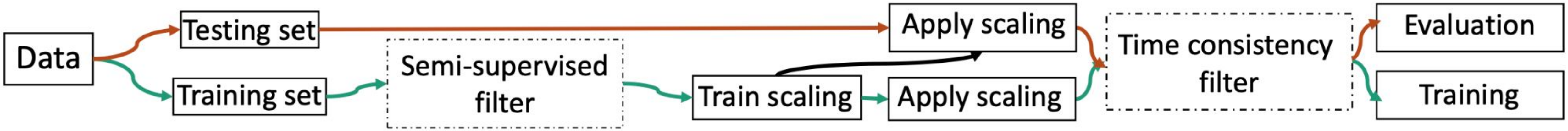
Borghesi et al., ExaTWIN'22

ExaMon: exascale ready monitoring framework for supercomputers

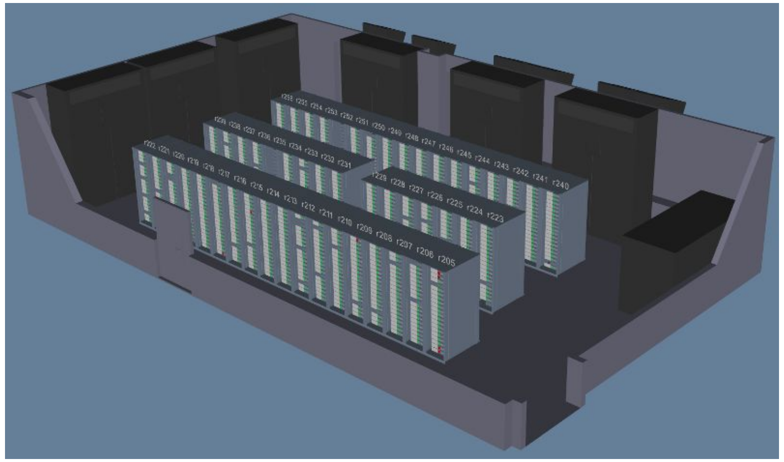


PRE-PROCESSING AND DATASET

Molan et al., RUAD'22

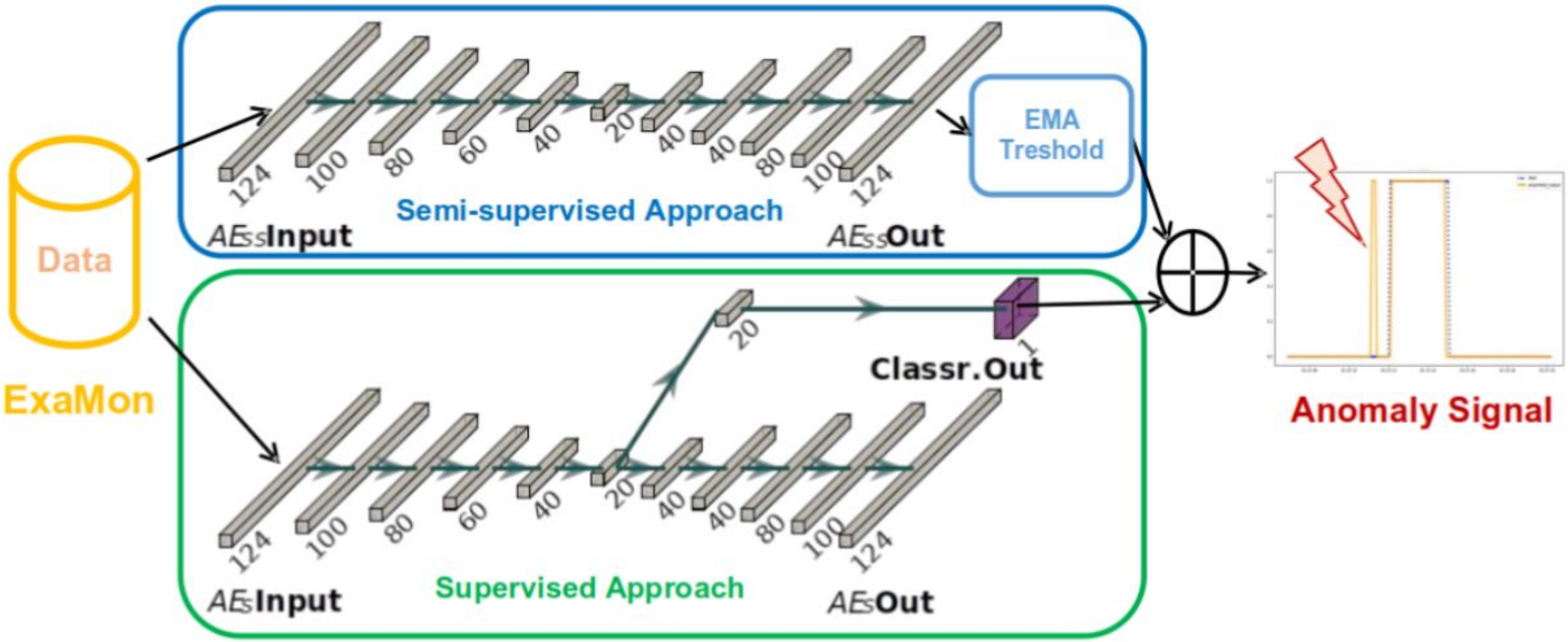


- Flexible pre-processing pipeline supporting semi-supervised and unsupervised training
- Dataset for the workshop is collected from Marconi 100



SoA FOR ANOMALY DETECTION

Borghesi et al., TPDS'22



SoA FOR ANOMALY DETECTION

Borghesi et al., TPDS'22

	Filters		
Model	Semi-supervised	Time consistency	Name
Trivial baseline: exponential smoothing	NO	YES	EXP
Unsupervised baseline: clustering	NO	NO	CLU
DENSE autoencoder baseline semi-supervised	YES	NO	$DENSE_{semi}$
DENSE autoencoder baseline unsupervised	NO	NO	$DENSE_{un}$
RUAD semi-supervised	YES	YES	$RUAD_{semi}$
RUAD unsupervised	NO	YES	$RUAD$

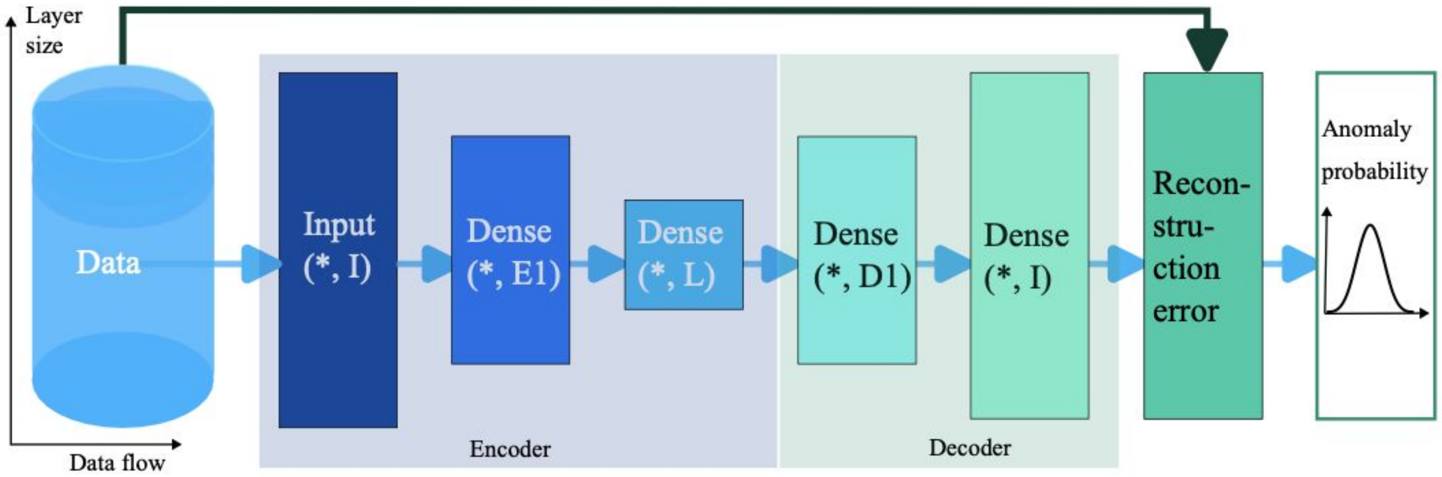
WHY UNSUPERVISED ANOMALY DETECTION?

Molan et al., RUAD'22

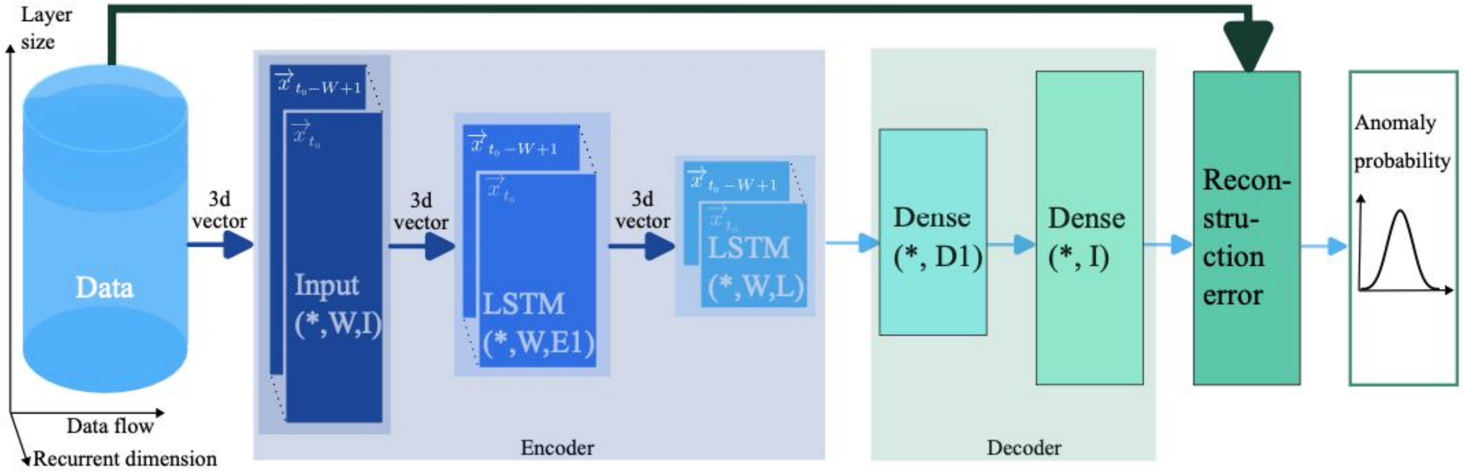
- Semi-supervised anomaly detection: models trained only on normal operation
- Need for (accurate) information about downtimes (anomaly timestamps)
- Difficult to deploy - accurate downtime information is **not** always available
- Motivation: train on **all data including anomalies**

Unsupervised anomaly detection: RUAD

Molan et al., RUAD'22



(a) Structure of baseline model - the dense autoencoder.



(b) Structure of the proposed RUAD model consisting of the LSTM encoder and dense decoder.

RESULTS ON A COMPLETE HPC SYSTEM MARCONI 100

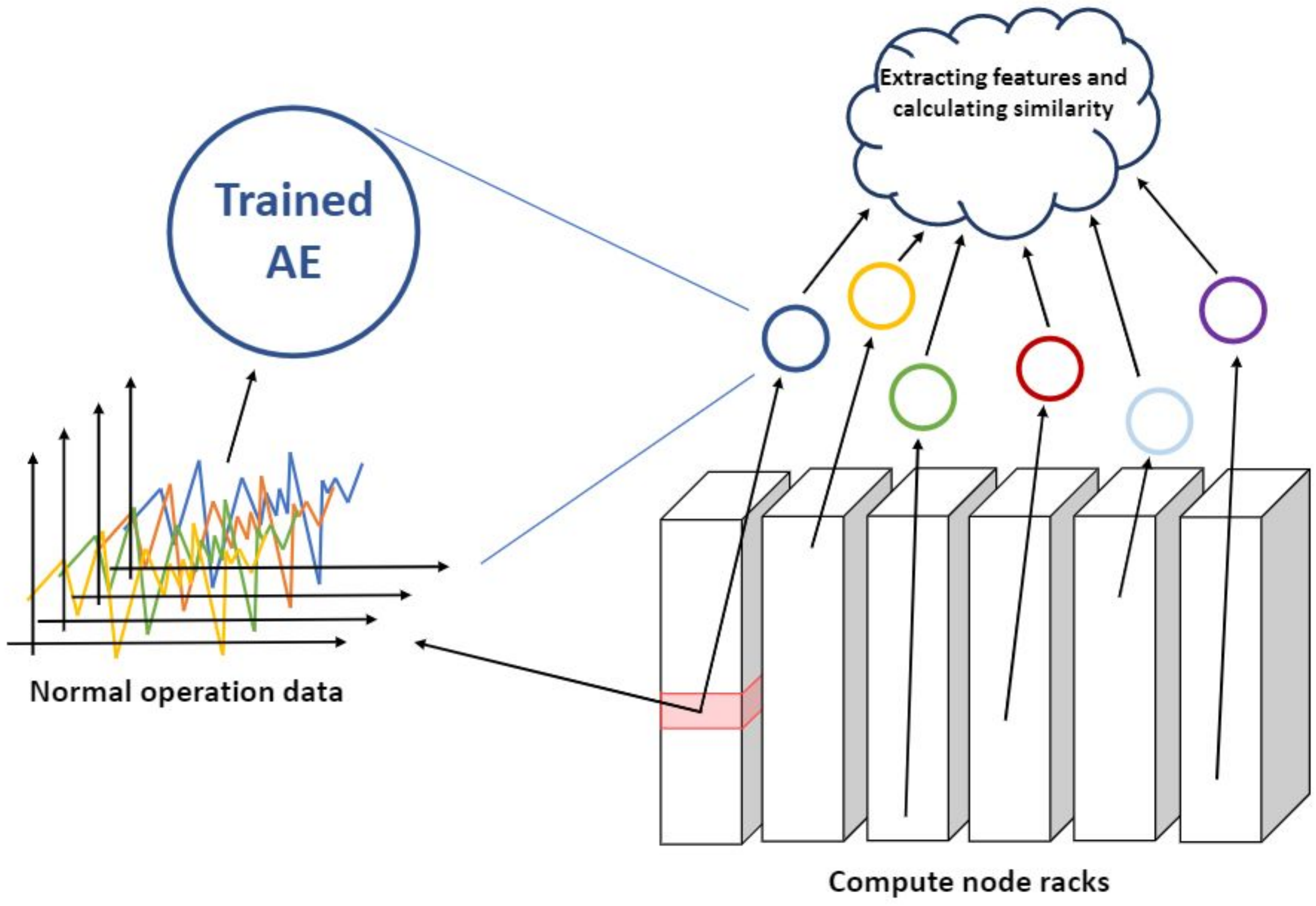
Molan et al., RUAD'22

Method	Combined ROC score
EXP	0.4276
CLU	0.5478
$DENSE_{semi}$	0.7470
$DENSE_{un}$	0.7344

Method	Combined ROC score			
Sequence length	5	10	20	40
$RUAD_{semi}$	0.7632	0.7582	0.7602	0.7446
$RUAD$	**0.7651**	**0.7672**	**0.7655**	**0.7473**

ANALYSING SUPERCOMPUTER NODES BEHAVIOUR WITH LATENT REPRESENTATIONS OF DEEP LEARNING MODELS

Molan et al., Euro-par'22



ANALYSING SUPERCOMPUTER NODES BEHAVIOUR WITH LATENT REPRESENTATIONS OF DEEP LEARNING MODELS

Molan et al., Euro-par'22

- Motivation: comparing populations by comparing fitted distributions
- Autoencoders are a good representation of node behaviour
- Overview of the approach:
  a. Separate model is trained for each node
  b. Features are extracted from nodes
  c. Based on extracted features similarity is calculated
  d. Similarity measure is used in hierarchical clustering

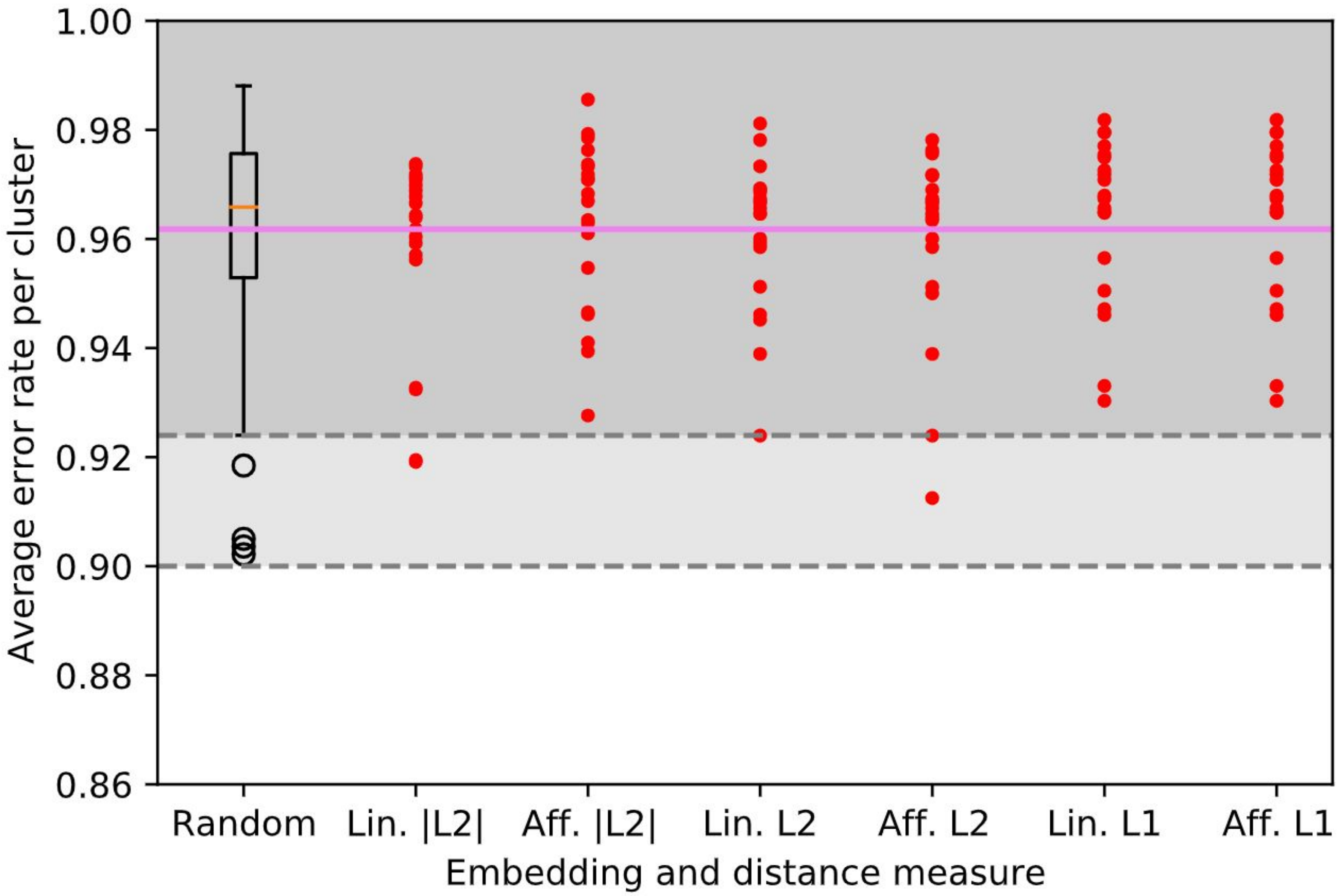
ANALYSING SUPERCOMPUTER NODES BEHAVIOUR WITH LATENT REPRESENTATIONS OF DEEP LEARNING MODELS

Molan et al., Euro-par'22

Distance measure:	Avg ava. in min. cluster:	Num. of nodes in min. cluster:
Sing. vector (Euc.)	0.9286	6
Vector of sing. values (Euc.)	0.8809	7
$W\vec{1} + \vec{b}$ (Euc.)	0.9126	8
$W\vec{1}$ (Euc.)	0.9367	5
W (absolute L2)	0.9191	7
A (absolute L2)	0.9276	5
W (L2)	0.9239	7
A (L2)	0.9124	10
W (L1)	0.9303	8
A (L1)	0.9303	8
Random sampling	0.9021	Not applicable

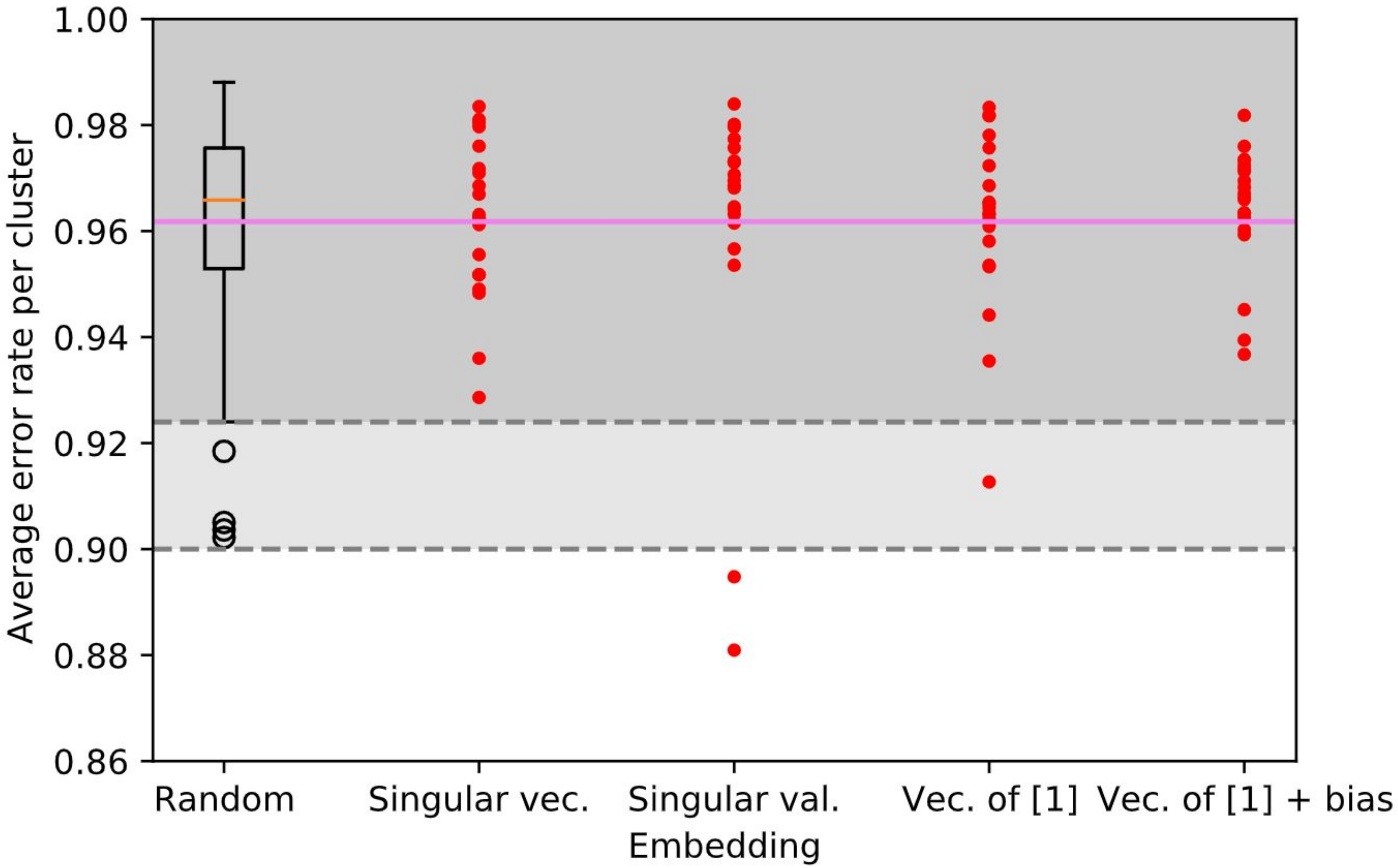
ANALYSING SUPERCOMPUTER NODES BEHAVIOUR WITH LATENT REPRESENTATIONS OF DEEP LEARNING MODELS

Molan et al., Euro-par'22



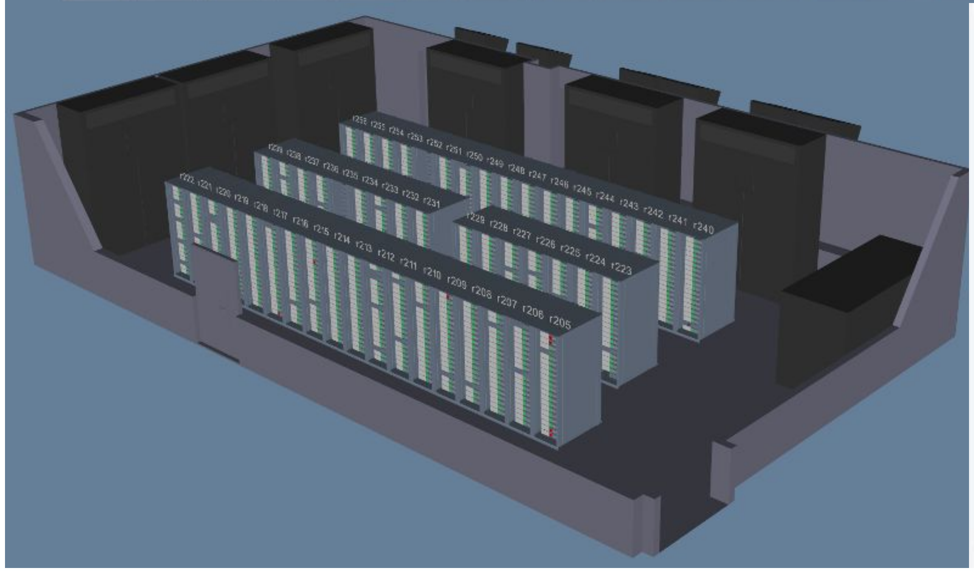
ANALYSING SUPERCOMPUTER NODES BEHAVIOUR WITH LATENT REPRESENTATIONS OF DEEP LEARNING MODELS

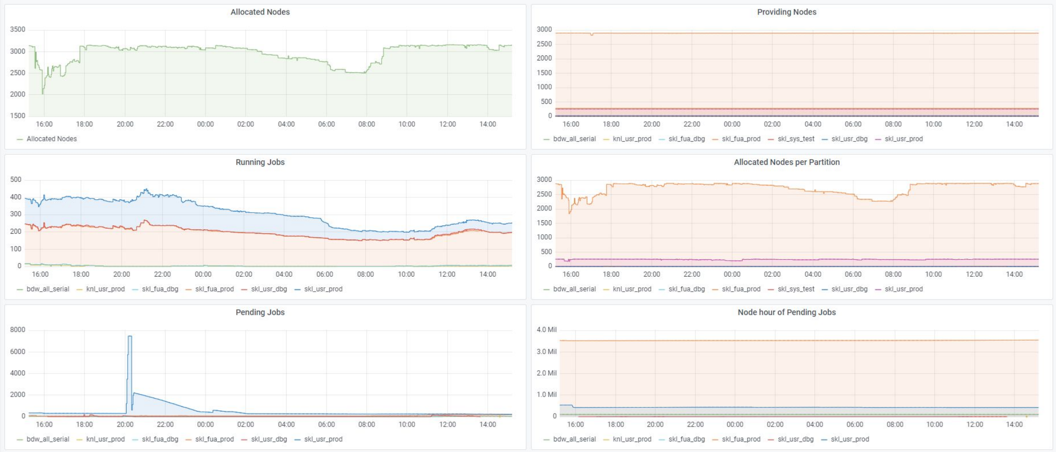
Molan et al., Euro-par'22

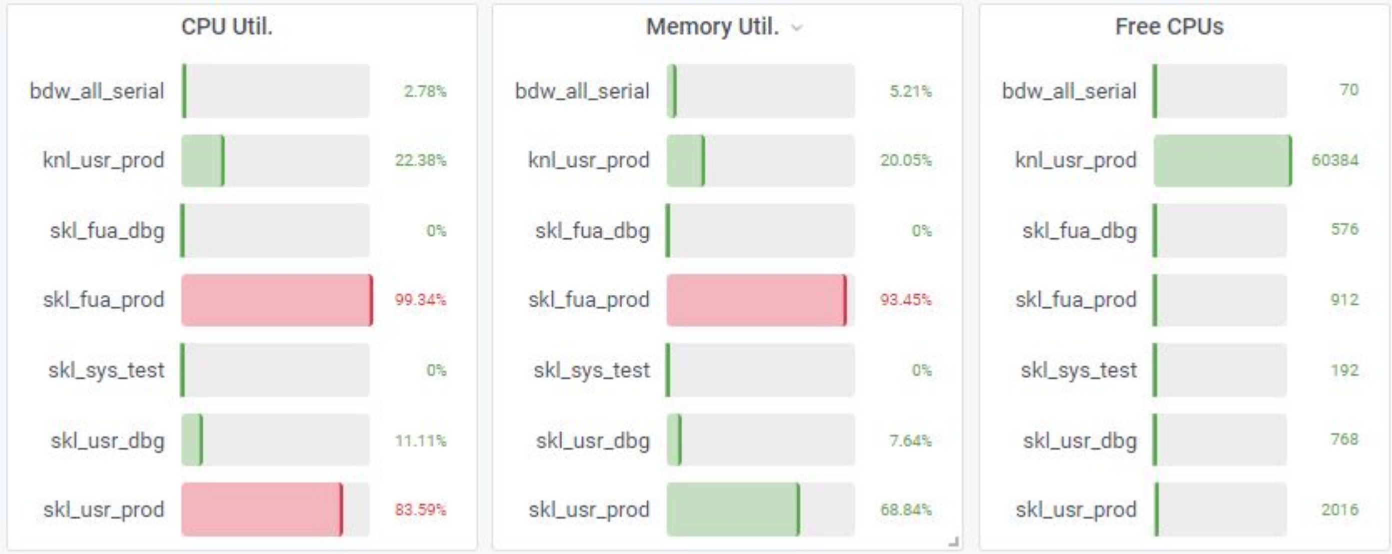


HPC DIGITAL TWIN

Borghesi et al., ExaTWIN'22





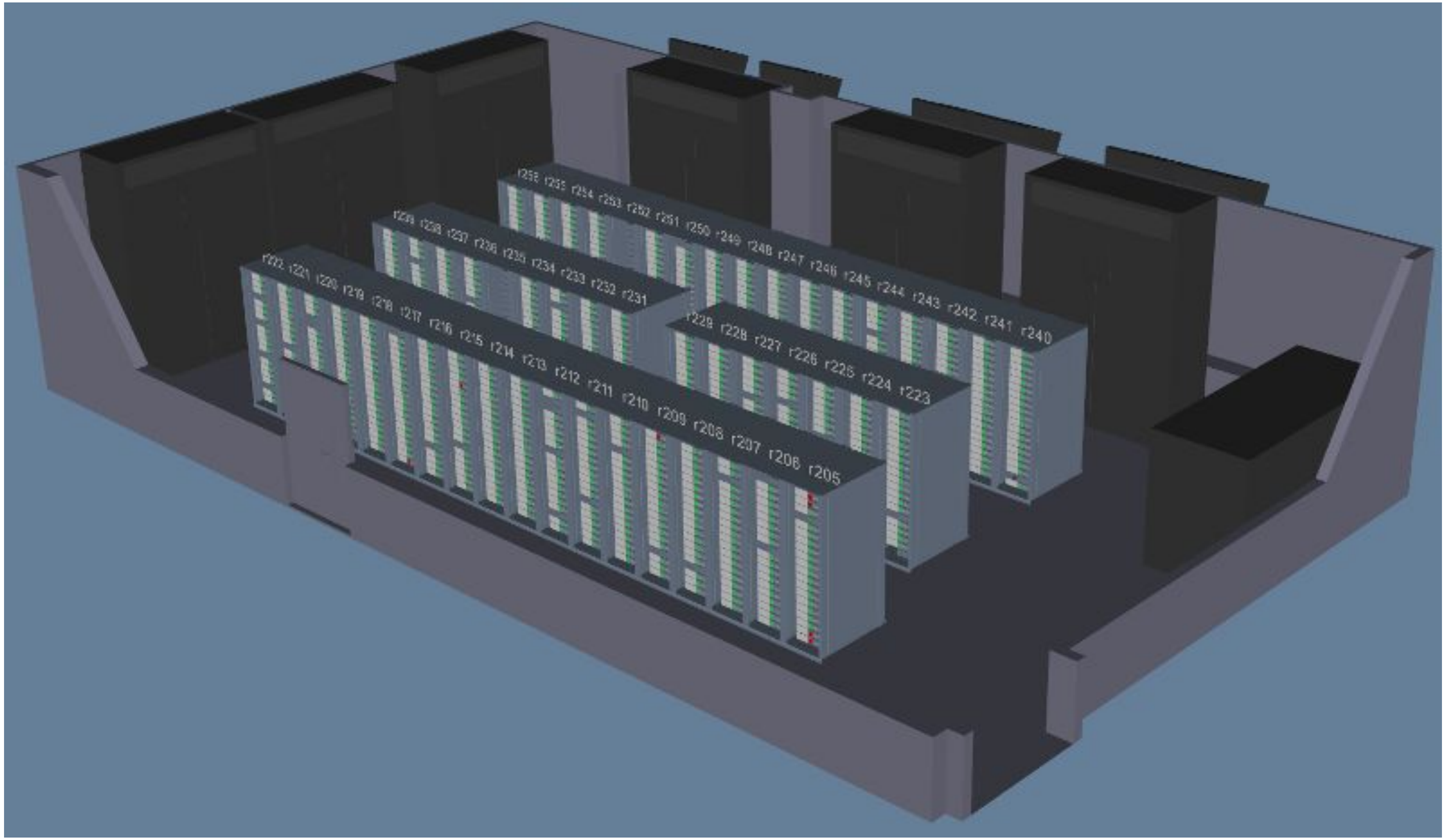


HPC DIGITAL TWIN

Borghesi et al., ExaTWIN'22

HPC DIGITAL TWIN

Borghesi et al., ExaTWIN'22

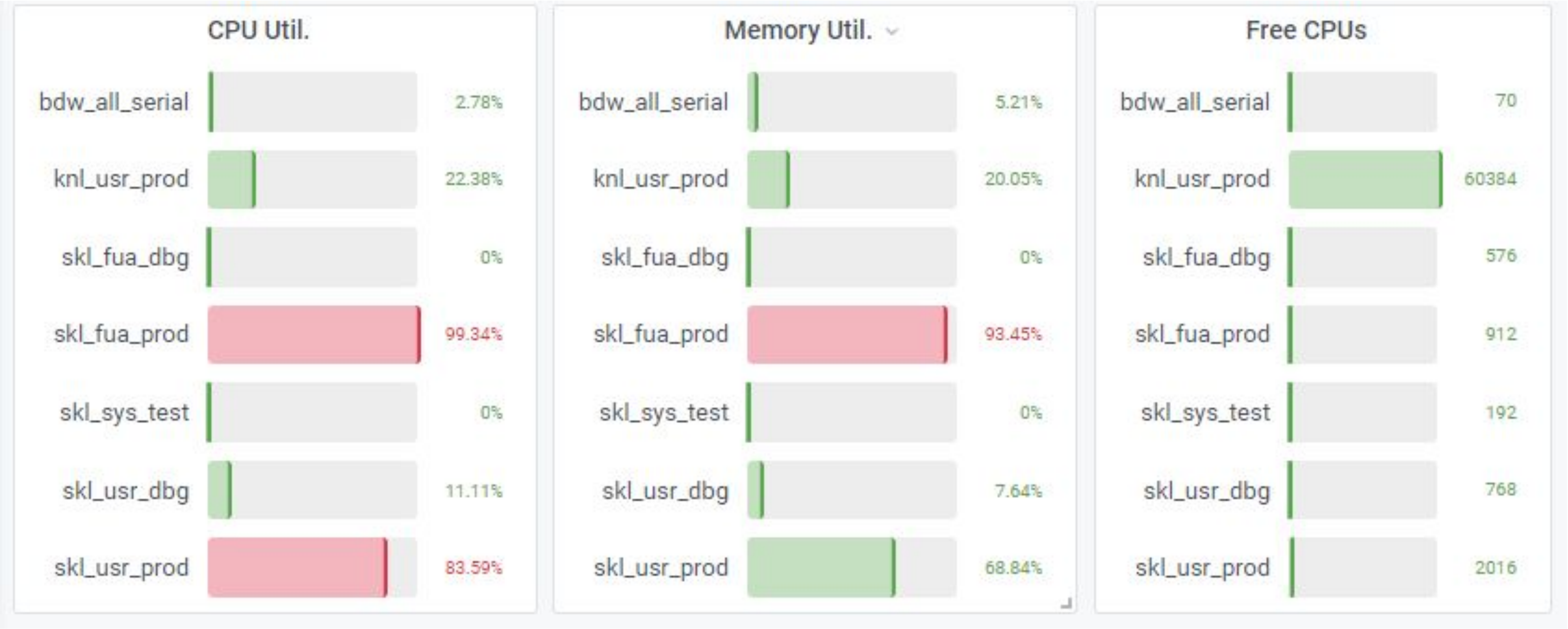


HPC DIGITAL TWIN

Borghesi et al., ExaTWIN'22

HPC DIGITAL TWIN

Borghesi et al., ExaTWIN'22



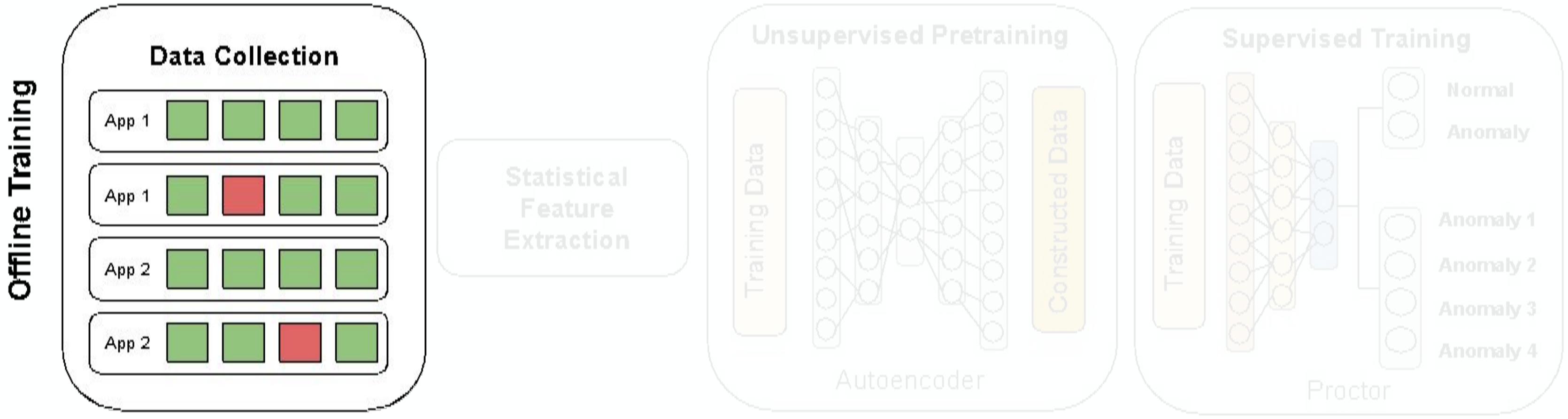
Aksar et al., ISC'21 - [Open Source]

PROCTOR:
A SEMI-SUPERVISED PERFORMANCE ANOMALY DIAGNOSIS FRAMEWORK

- A semi-supervised framework to detect and diagnose performance anomalies
  - Significantly less labeled data compared to baselines
- Evaluation on a production HPC system and a testbed HPC cluster
  - 11% better F-score on average

Aksar et al., ISC'21 - [Open Source]

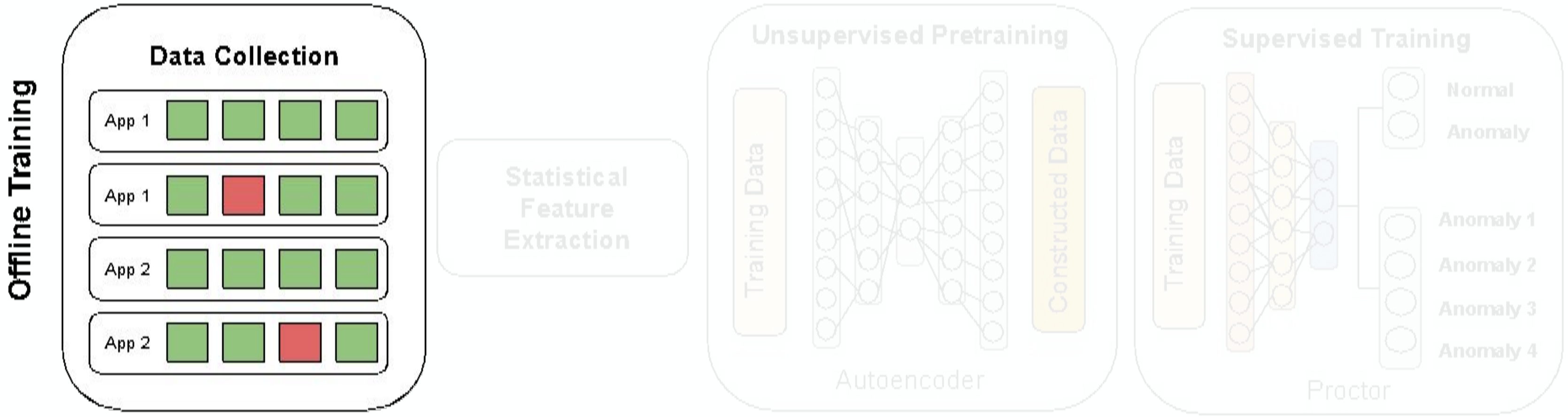
PROCTOR: MONITORING



- Run synthetic anomalies with different real and proxy HPC applications
  - Anomalies mimic common performance variations

Aksar et al., ISC'21 - [Open Source]

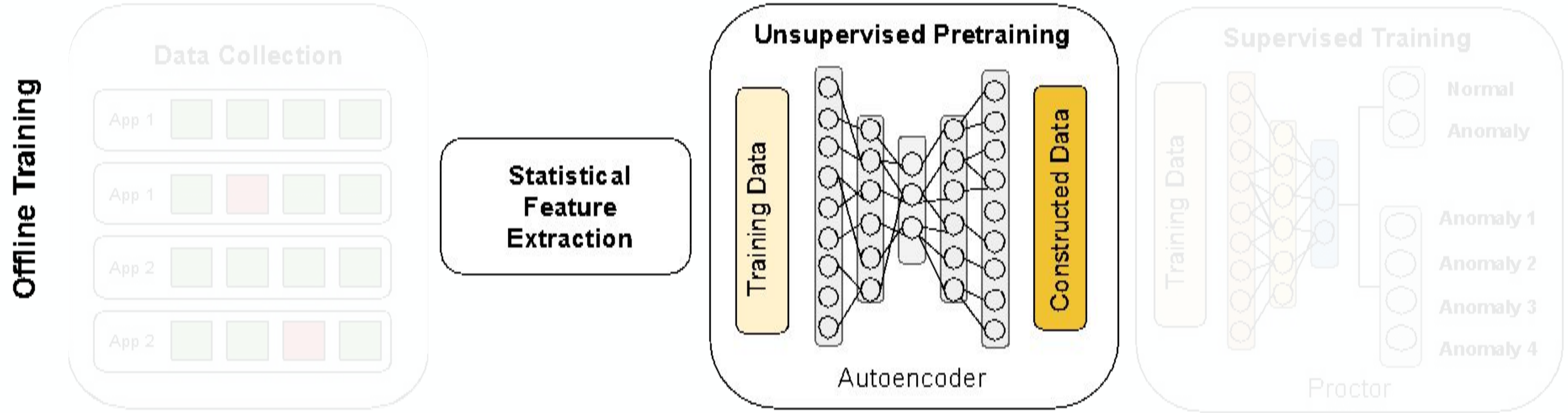
PROCTOR: MONITORING



- Collect telemetry data using Lightweight Distributed Metric System (LDMS) *[Agelastos et al., SC'12]*
  - 100s of time series per node
  - Hardware counters, memory/CPU usage, etc.

Aksar et al., ISC'21 - [Open Source]

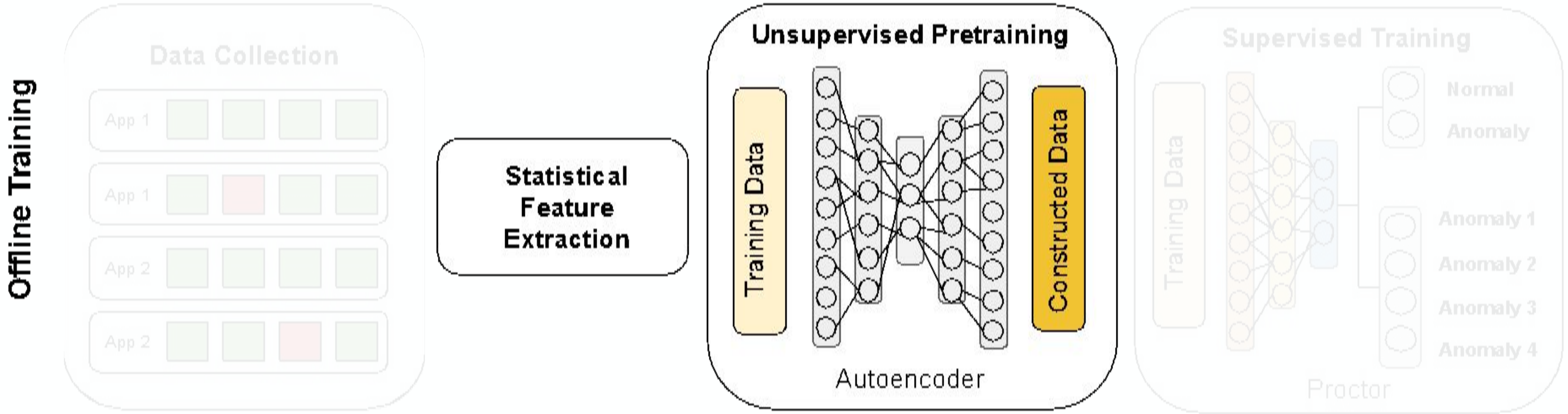
PROCTOR: AUTOENCODER TRAINING



- Extract statistical features that retain the raw time series' characteristics
  - Remove application initialization and finalization periods
  - Transform cumulative counters into events/sec

Aksar et al., ISC'21 - [Open Source]

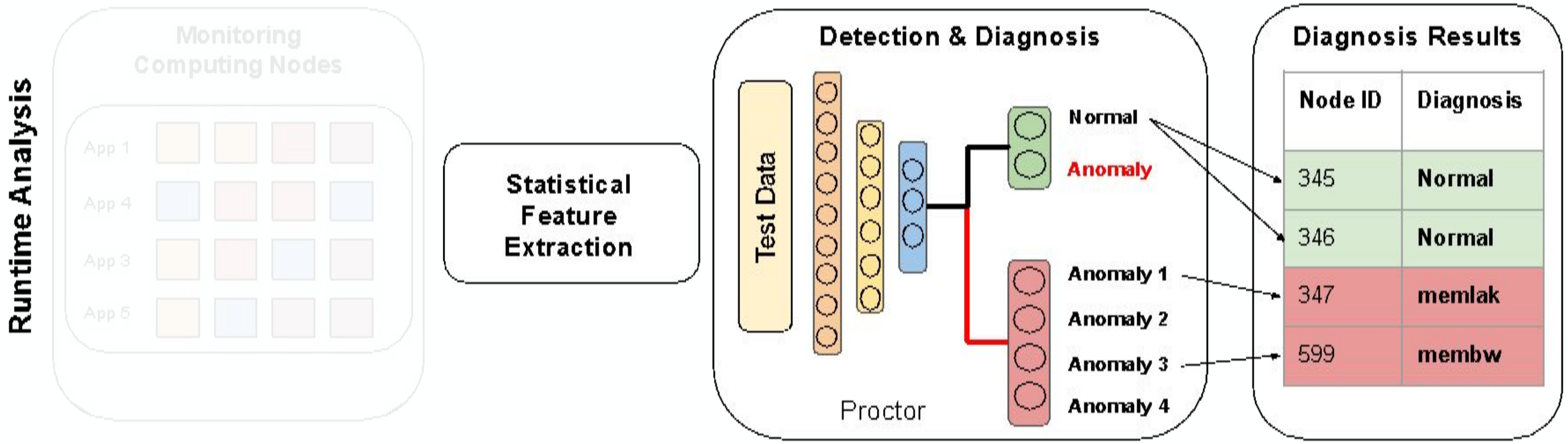
PROCTOR: AUTOENCODER TRAINING



- Autoencoder learns the representation of normal and anomalous runs in an unsupervised manner

Aksar et al., [ISC'21] - [Open Source]

PROCTOR: DIAGNOSIS



- Use the trained autoencoder's encoder and perform *two-level* classification using a few labeled samples
  - First classifier learns to classify anomalous vs. normal
  - Second classifier learns to classify the type of the anomalies

Aksar et al., ISC'21 - [Open Source]

EXPERIMENTAL METHODOLOGY – BASELINE METHODS

- RF-Tuncer *[Tuncer et al., TPDS'18]*
  - Statistical feature extraction and feature selection to train decision tree-based models
  - Anomaly diagnosis
- AE-Borghesi *[Borghesi et al., EAAI'19]*
  - Autoencoder trained on only *normal* samples and selects a threshold
  - Anomaly detection

Aksar et al., ISC'21 - [Open Source]

EXPERIMENTAL METHODOLOGY – EVALUATION

- ***F1-score***
  - The harmonic mean of precision and recall
- ***False alarm rate***
  - Classifying a normal sample as any type of anomaly
- ***Anomaly miss rate***
  - Classifying any of the anomalous samples as normal

$$False\ Alarm\ Rate = \frac{False\ Positives}{False\ Positives + True\ Negatives}$$

$$Anomaly\ Miss\ Rate = \frac{False\ Negatives}{False\ Negatives + True\ Positives}$$

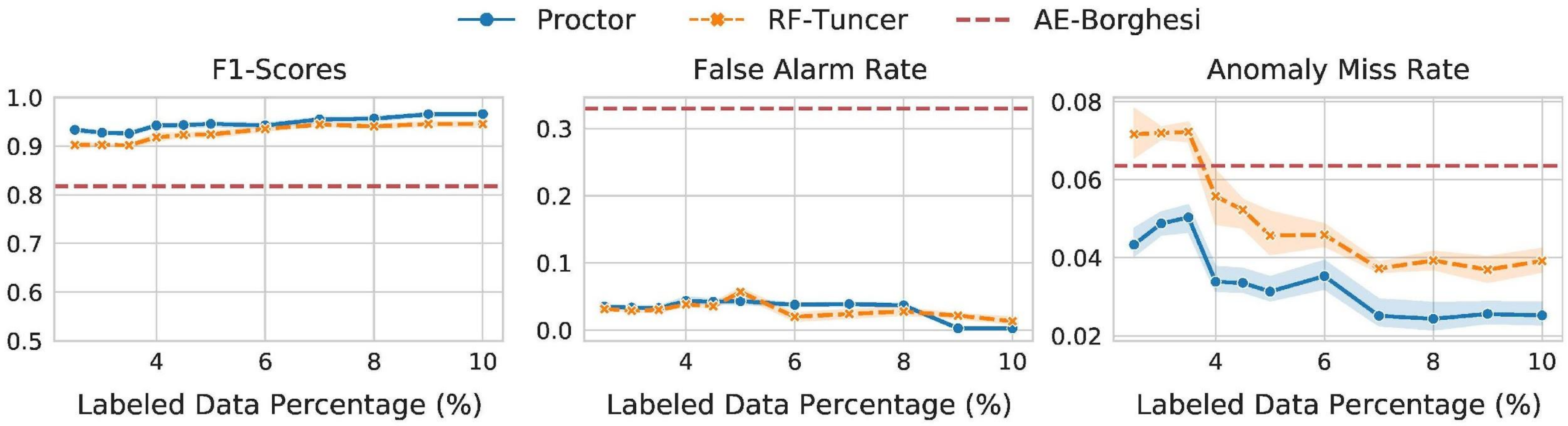
Aksar et al., ISC'21 - [Open Source]

DATASET PREPARATION

- Sample: Telemetry data collected during an application run from a compute node
- Eclipse:
  - 1526 normal samples and 2304 anomalous samples
  - Unlabeled Training Data: 611 normal & 68 anomalous
- Volta:
  - 18980 normal samples and 1932 anomalous samples
  - Unlabeled Training Data: 5694 normal & 618 anomalous
- Labeled training data - only for Proctor and RF-Tuncer
  - Eclipse: 2%, 3%, 4%, 5%, 6%, 8%, 10% of unsupervised training data
  - Volta : 0.1%, 0.15%, 0.2%, 0.25%, 0.30%, 0.35% of unsupervised training data

Aksar et al., ISC'21 - [Open Source]

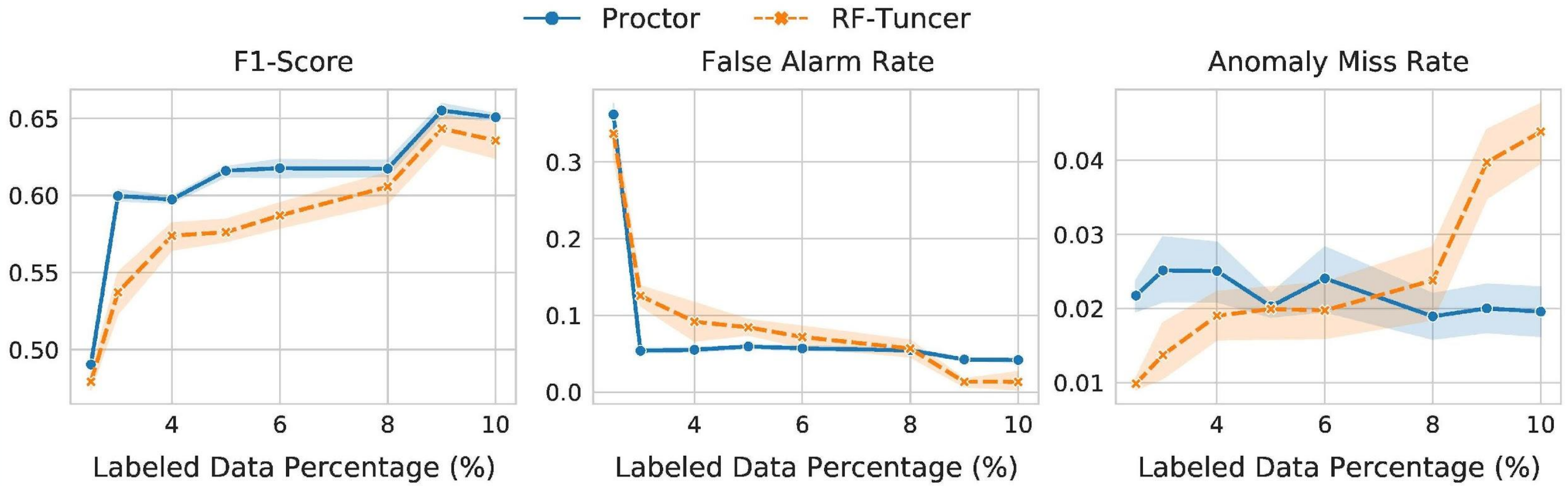
EVALUATION – ANOMALY DETECTION (ECLIPSE)



- Proctor outperforms the baselines in F1-score and anomaly miss rate
- Proctor maintains a similar performance with RF-Tuncer in false alarm rate

Aksar et al., ISC'21 - [Open Source]

EVALUATION – ANOMALY DIAGNOSIS (ECLIPSE)



- Proctor outperforms RF-Tuncer by 4.5% on average in F1-score
- Maintains very low false alarm rate and anomaly miss rate

AGENDA

- Overview: Telemetry Data-based Analytics on Large-scale Computing Systems
- Supervised Methods: Anomaly Detection and Diagnosis
- Break
- Semi-supervised and Unsupervised Anomaly Detection in Supercomputers
- **Deployment: Challenges and Current Status**
- Hands-on Activity

DEPLOYMENT PERSPECTIVE - RELATED WORK

- Deploying node-specific and node-agnostic anomaly detection models to a small-scale cluster *[Borghesi et al., EAAI'19]*
- Deploying models to forecast node power and identify abnormal behaviors on a cluster *[Netti et al., HPDC'20]*
- Deploying a supervised ML framework to a production system *[Aksar et al., Euro-par'21]*

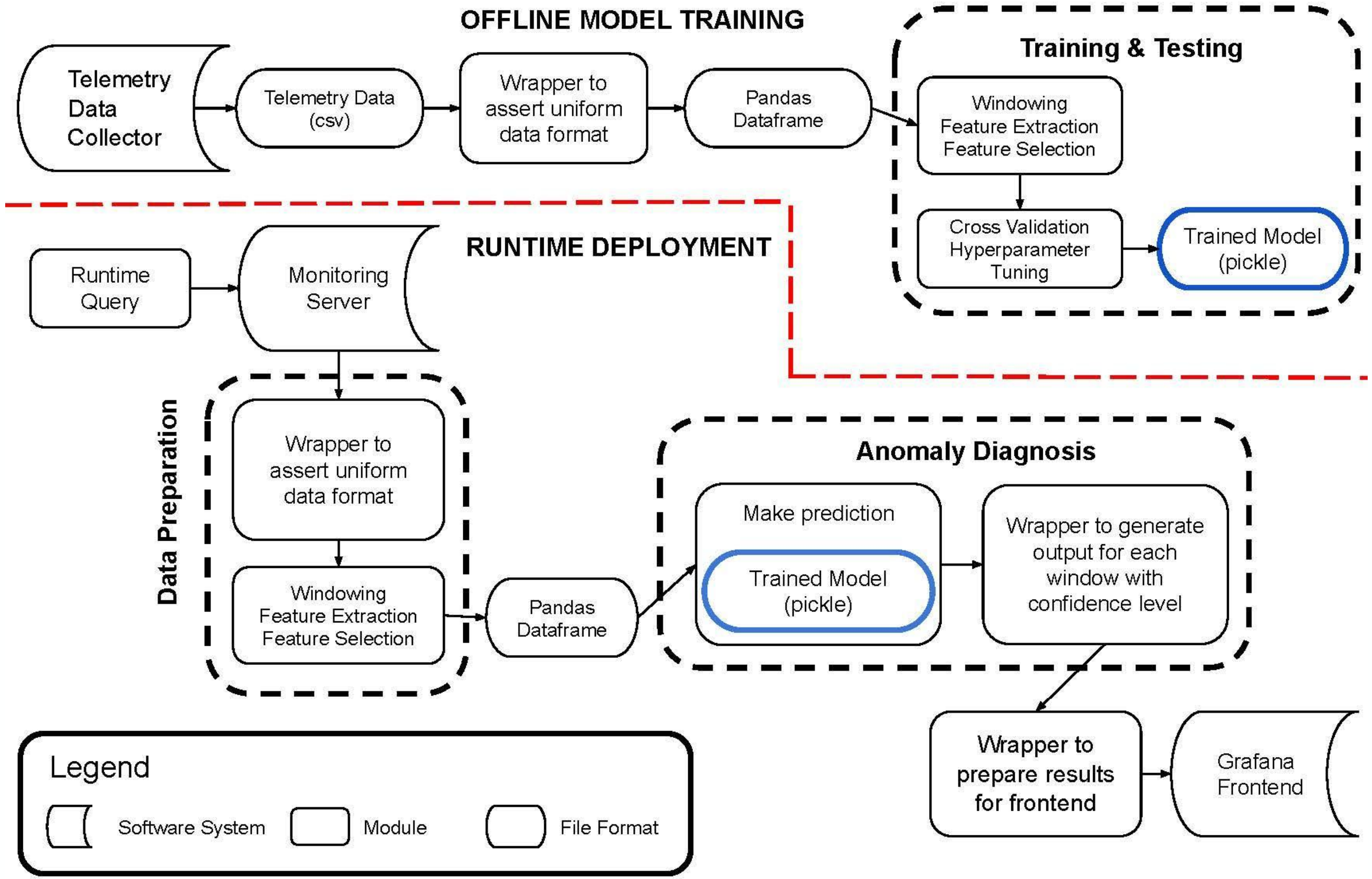
Aksar et al., Euro-Par'21 - [Open Source]

E2EWATCH:
AN END-TO-END ANOMALY DIAGNOSIS FRAMEWORK FOR PRODUCTION HPC SYSTEMS

- Deployment of an end-to-end anomaly diagnosis framework on a 1488-node production HPC system
  - Job and node-level analysis
  - Deliver results in near-real time
  - Customizable and interpretable visualization

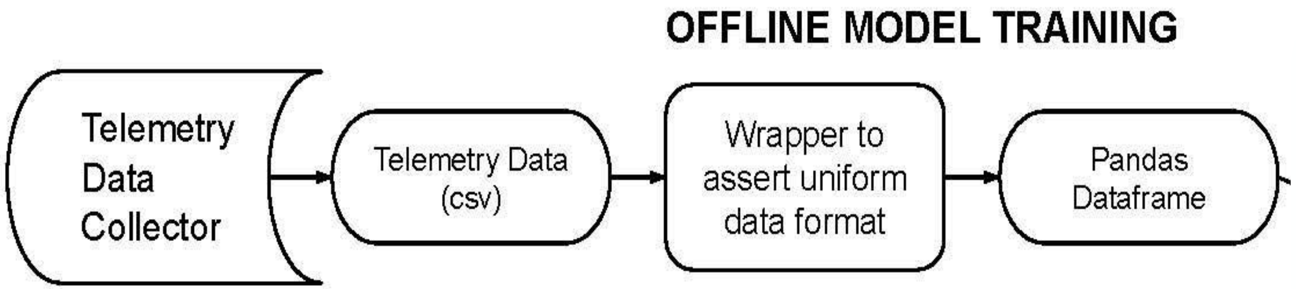
Aksar et al., Euro-Par'21 - [Open Source]

E2EWATCH: SUMMARY



Aksar et al., Euro-Par'21 - [Open Source]

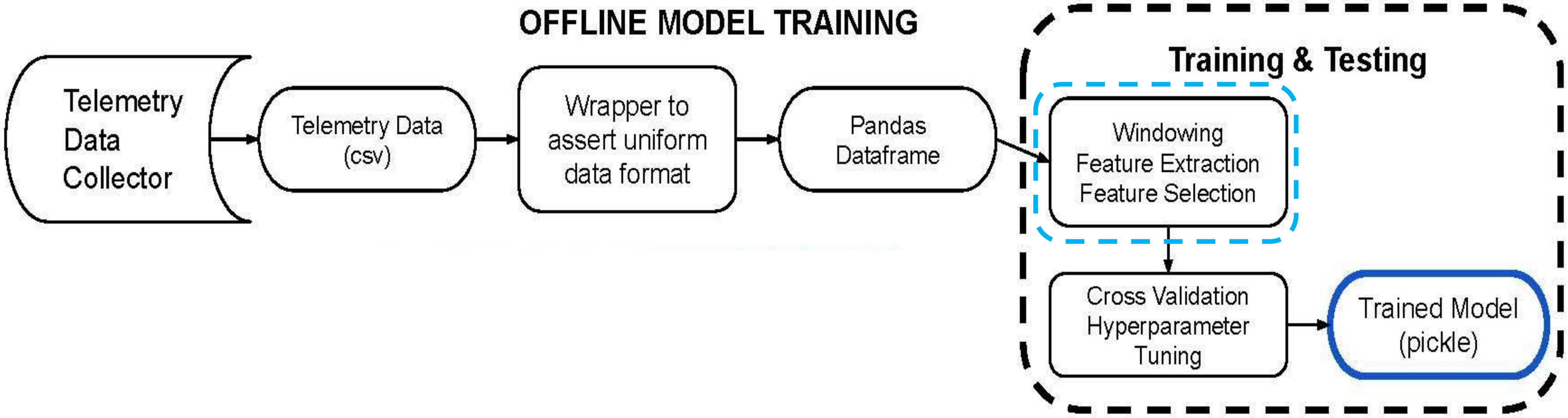
E2EWATCH: MONITORING



- Collect telemetry data during controlled experiments with and without synthetic anomalies

Aksar et al., Euro-Par'21 - [Open Source]

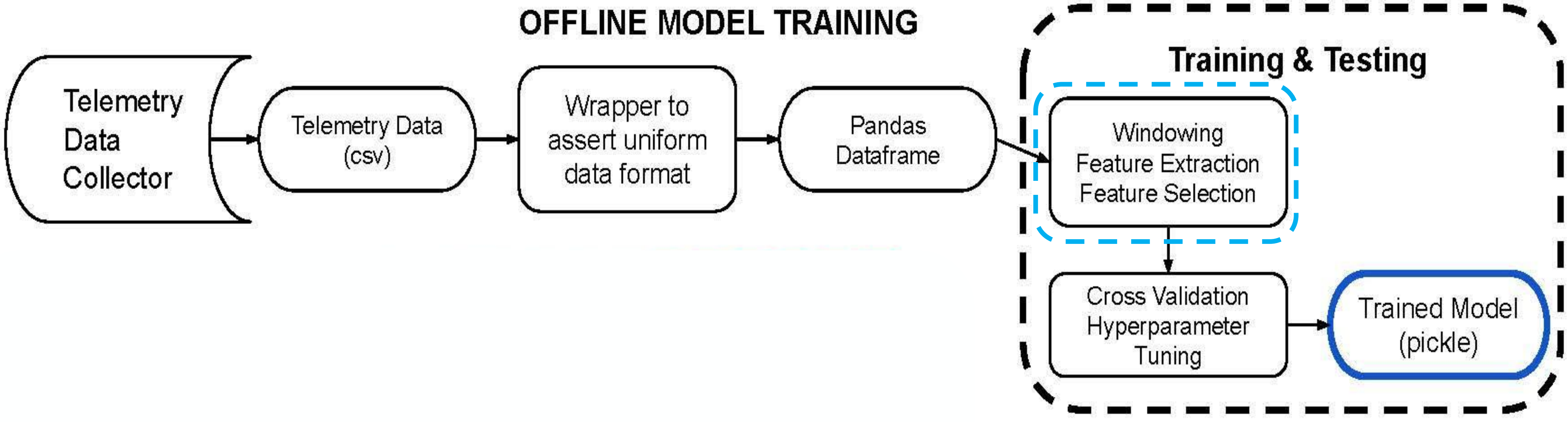
E2EWATCH: DATA PREPARATION



- Divide raw time series into multiple equal-length overlapping windows with 15-seconds skip intervals
  - E.g., [0-45], [15-60]

Aksar et al., Euro-Par'21 - [Open Source]

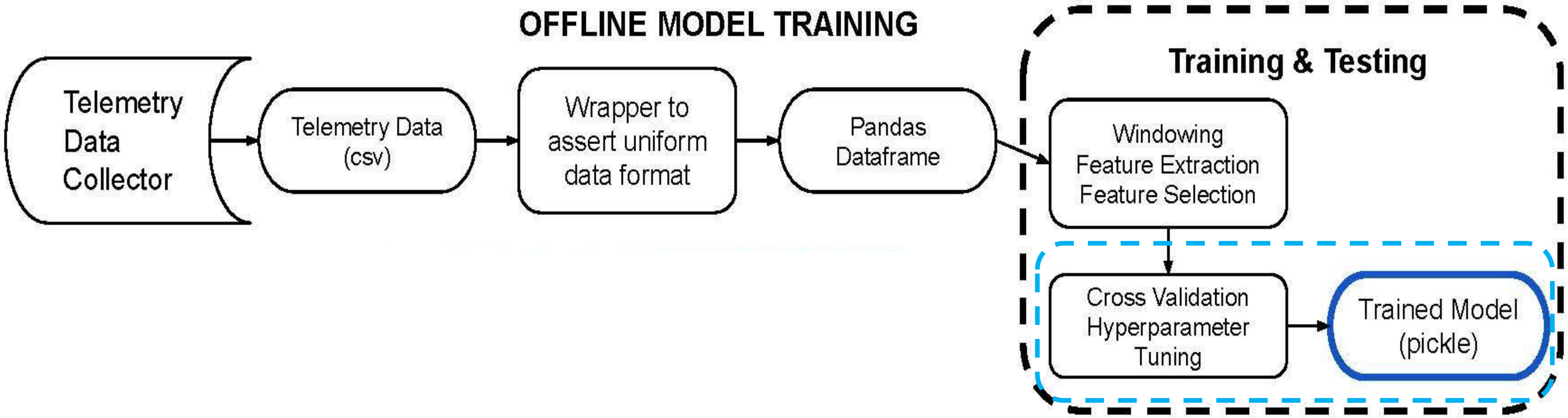
E2EWATCH: DATA PREPARATION



- Calculate the following statistical features of each window
  - Minimum; maximum; 5^{th} , 25^{th} , 50^{th} , 75^{th} , 95^{th} percentiles; mean; variance; skewness

Aksar et al., Euro-Par'21 - [Open Source]

E2EWATCH: OFFLINE MODEL TRAINING

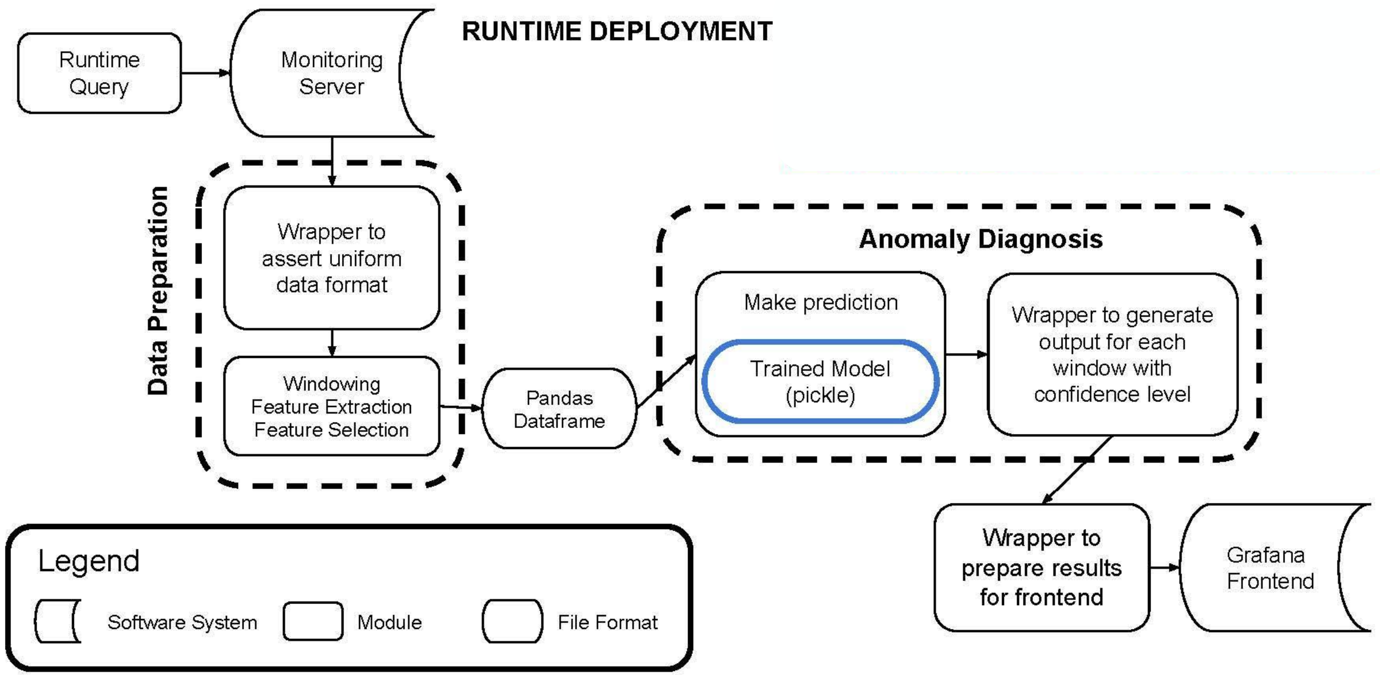


- Hyperparameter tuning and K-fold cross validation
- Select the best performing model
- Store model as *pickle* in the monitoring server

Aksar et al., Euro-Par'21 - [Open Source]

E2EWATCH: RUNTIME DEPLOYMENT

- The same data preparation phase
- Use the pickled model to make predictions
- Send results to Grafana user interface



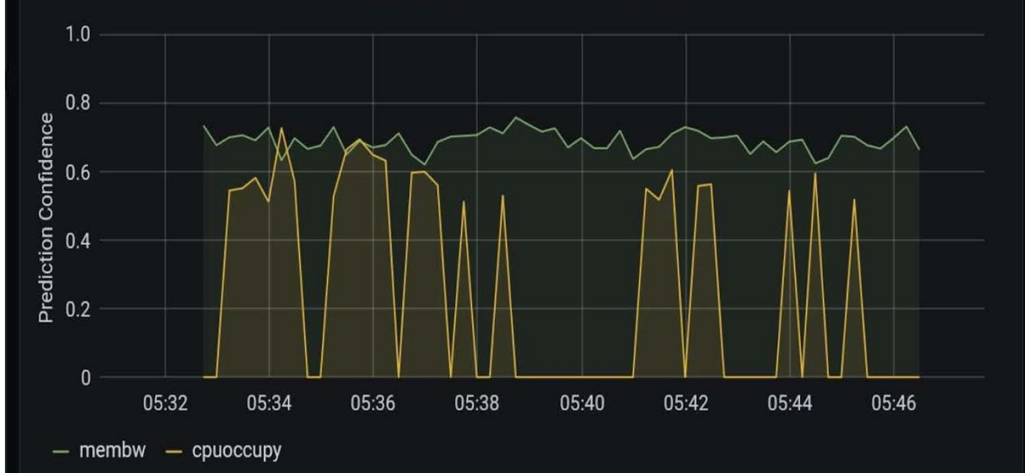
Aksar et al., Euro-Par'21 - [Open Source]

E2EWATCH: FRONTEND

Job Level Breakdown of Anomaly Detection

Anomaly Type	% of nodes and timesteps with Anomaly Type	Prediction Confidence
cpuoccupy	8.48	medium
membw	91.52	medium

Time Series of Prediction Confidence



Node Level Breakdown of Anomaly Detection

Node ID	Anomaly Type	% of timesteps with Anomaly Type	Prediction Confidence
0	membw	98.21	medium
0	cpuoccupy	1.79	medium
1	membw	78.57	medium
1	cpuoccupy	21.43	low
2	membw	100.00	high
3	membw	80.36	medium
3	cpuoccupy	19.64	medium
4	membw	98.21	medium
4	cpuoccupy	1.79	medium
5	membw	92.86	medium
5	cpuoccupy	7.14	medium
6	membw	85.71	medium
6	cpuoccupy	14.29	medium
7	membw	98.21	medium
7	cpuoccupy	1.79	medium

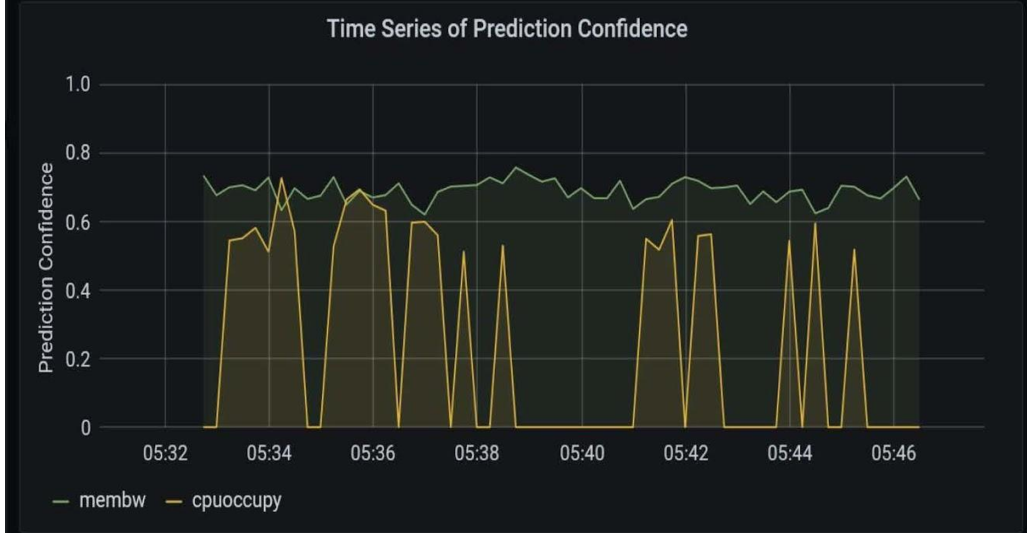
Orange box shows detected anomaly types during selected job

Aksar et al., Euro-Par'21 - [Open Source]

E2EWATCH: FRONTEND

Job Level Breakdown of Anomaly Detection

Anomaly Type	% of nodes and timesteps with Anomaly Type	Prediction Confidence
cpuoccupy	8.48	medium
membw	91.52	medium



Node Level Breakdown of Anomaly Detection

Node ID	Anomaly Type	% of timesteps with Anomaly Type	Prediction Confidence
0	membw	98.21	medium
0	cpuoccupy	1.79	medium
1	membw	78.57	medium
1	cpuoccupy	21.43	low
2	membw	100.00	high
3	membw	80.36	medium
3	cpuoccupy	19.64	medium
4	membw	98.21	medium
4	cpuoccupy	1.79	medium
5	membw	92.86	medium
5	cpuoccupy	7.14	medium
6	membw	85.71	medium
6	cpuoccupy	14.29	medium
7	membw	98.21	medium
7	cpuoccupy	1.79	medium

Yellow box shows anomaly percentages across all computed windows
Green box shows prediction confidences

Aksar et al., Euro-Par’21 - [Open Source]

E2EWATCH: FRONTEND

Job Level Breakdown of Anomaly Detection

Anomaly Type	% of nodes and timesteps with Anomaly Type	Prediction Confidence
cpuoccupy	8.48	medium
membw	91.52	medium

Node Level Breakdown of Anomaly Detection

Node ID	Anomaly Type	% of timesteps with Anomaly Type	Prediction Confidence
0	membw	98.21	medium
0	cpuoccupy	1.79	medium
1	membw	78.57	medium
1	cpuoccupy	21.43	low
2	membw	100.00	high
3	membw	80.36	medium
3	cpuoccupy	19.64	medium
4	membw	98.21	medium
4	cpuoccupy	1.79	medium
5	membw	92.86	medium
5	cpuoccupy	7.14	medium
6	membw	85.71	medium
6	cpuoccupy	14.29	medium
7	membw	98.21	medium
7	cpuoccupy	1.79	medium

Red box shows node-level breakdown for the selected job id

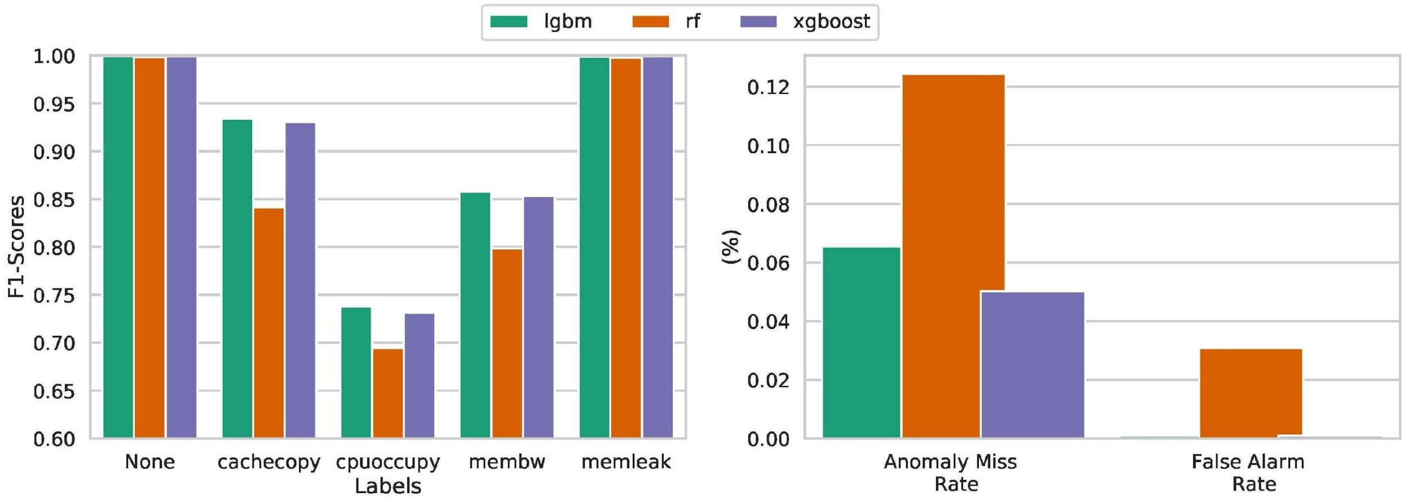
Aksar et al., Euro-Par'21 - [Open Source]

EXPERIMENTAL METHODOLOGY – MODELS

- *Extreme Gradient Boosting (XGBoost)*
  - Uses gradient boosting which is an ensemble of weak learners
- *Light Gradient Boosting Machine (LGBM)*
  - Similar to XGBoost but it has different node splitting
- *Random Forest (RF)*
  - Combines results of multiple decision trees

Aksar et al., Euro-Par'21 - [Open Source]

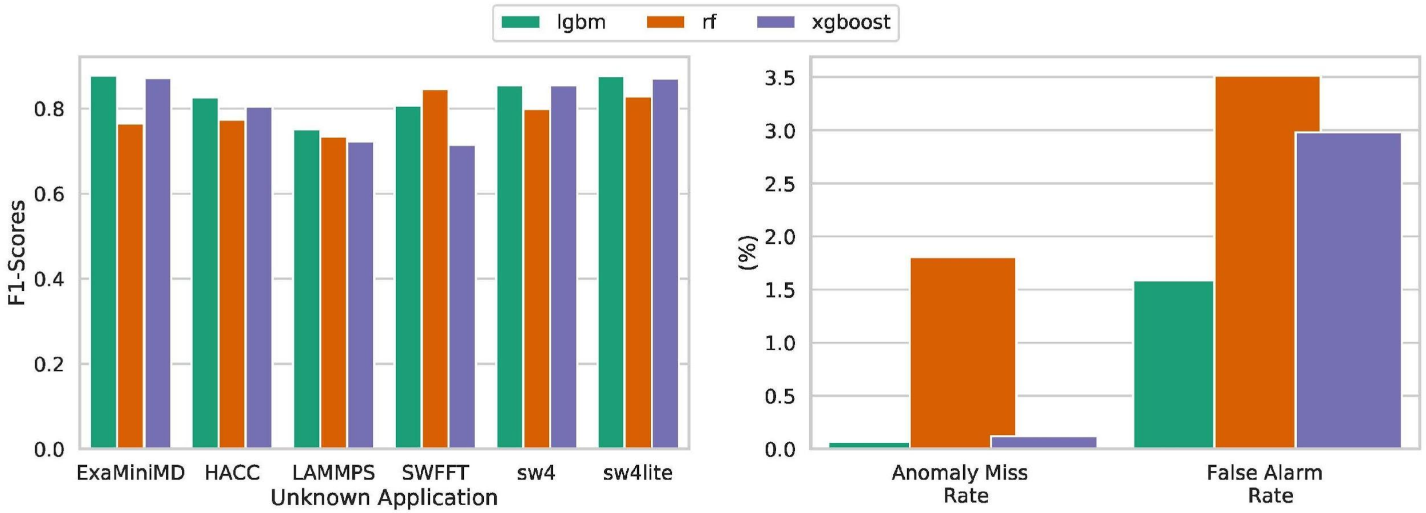
EVALUATION – ANOMALY DIAGNOSIS



- Almost perfect diagnosis in without anomaly case
- LGBM and XGBoost perform up to 10% better than RF
- *Cpuoccupy* is being confused with *membw* due to similar CPU utilization characteristics

Aksar et al., Euro-Par'21 - [Open Source]

EVALUATION – UNKNOWN APPLICATIONS



- Goal: Evaluate each model's performance when test data has unknown applications
  - Remove all runs of the selected application from the training set
  - Include only the removed application to the test set
- Except LAMMPS and SWFFT, XGBoost and LGBM perform up to 10% better than RF
- LGBM is the best considering anomaly miss rate and false alarm rate

DEPLOYMENT CHALLENGES

- Computation-heavy runtime analysis
  - Models are only run when a user requests and this saves significant energy
- Delivering results in near-real time
  - The database is specifically designed for the scale and HPC telemetry data
- Data transformation
  - Transforming the raw monitoring data into a format suitable for ML models
  - Transforming the results into a format for visualization
- Exa-scale scaling
  - How to train and deploy a ML framework that will handle thousands of compute nodes

AGENDA

- Overview: Telemetry Data-based Analytics on Large-scale Computing Systems
- Supervised Methods: Anomaly Detection and Diagnosis
- Break
- Semi-supervised and Unsupervised Anomaly Detection in Supercomputers
- Deployment: Challenges and Current Status
- **Hands-on Activity**

HANDS-ON ACTIVITY

Github Repository: https://github.com/MolanM/Hands-on

ACKNOWLEDGMENTS

- This research was partly supported by the EuroHPC EU PILOT project (g.a.101034126), the EuroHPC EU Regale project (g.a. 956560), EU H2020-ICT-11-2018-2019 IoTwins project (g.a. 857191), and EU Pilot for exascale EuroHPC EUPEX (g. a. 101033975).

- This work has been partially funded by Sandia National Laboratories. Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and 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. This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

COLLABORATORS

- Jim Brandt - Sandia National Labs
- Vitus J. Leung - Sandia National Labs
- Benjamin Schwaller - Sandia National Labs
- Omar Aaziz - Sandia National Labs
- Brian Kulis - Boston University
- Manuel Egele - Boston University