Investigating Anomalies in Compute Clusters: An Unsupervised Learning Approach

Yiyang Lu†, Jie Ren†, Yasir Alanazi§, Ahmed Mohammed§, Diana McSpadden§, Laura Hild§, Mark Jones§, Wesley Moore§, Malachi Schram§, Bryan Hess§, Evgenia Smirni†

†College of William and Mary §Thomas Jefferson National Accelerator Facility

Motivation

- Managing cluster-level anomalies, even at smaller scales, is complex due to interconnected jobs and infrastructure.
- Compute clusters benefit from the **timely detection of anomalous events** and **detailed root cause analysis**.
  - ✓ Enable preemptive detection of hardware failures.
  - ✓ Enable proactive job redistribution to prevent job progress loss due to system anomalies.
  - ✓ Alleviate the system administrator's burden in system maintenance and reduces the time required for anomaly resolution.

Challenges

- Labelling anomalies for model training is impractical in compute clusters.
  - Only 0.035% anomalies in all monitored data in real-world HPC cluster[1].
- Large amount of monitored metrics and multiple possible sources of anomalies.
  - Due to a lack of understanding of which monitored metrics are significant in identifying anomalies, it's hard to choose appropriate metrics for detection.
- Hardware heterogeneity increases complicity of compute cluster management.
  - Anomaly detection on nodes with different hardware properties.

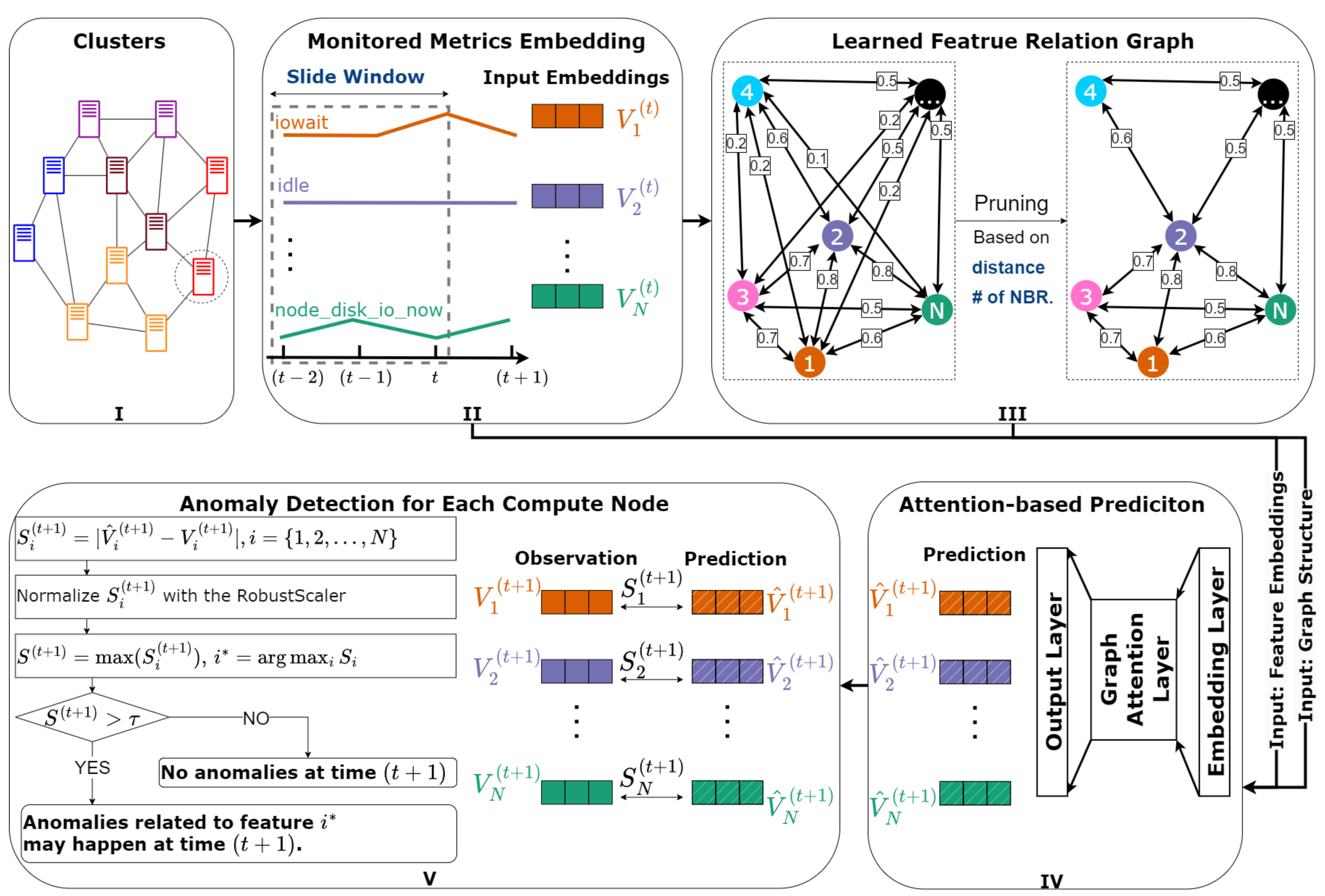
Attention-based Graph Neural Network

Goal:

- Compute node level anomaly prediction.
- Hardware component level root cause analysis.

Method:

- Unsupervised anomaly detection: only normal events collected for training.
- Learn the relations among the N monitored metrics within one compute node.
- Identify anomalies with significant deviations of predicted future time series from expected behavior for each monitored metrics.



Workflow of using attention-based GNN for anomaly prediction

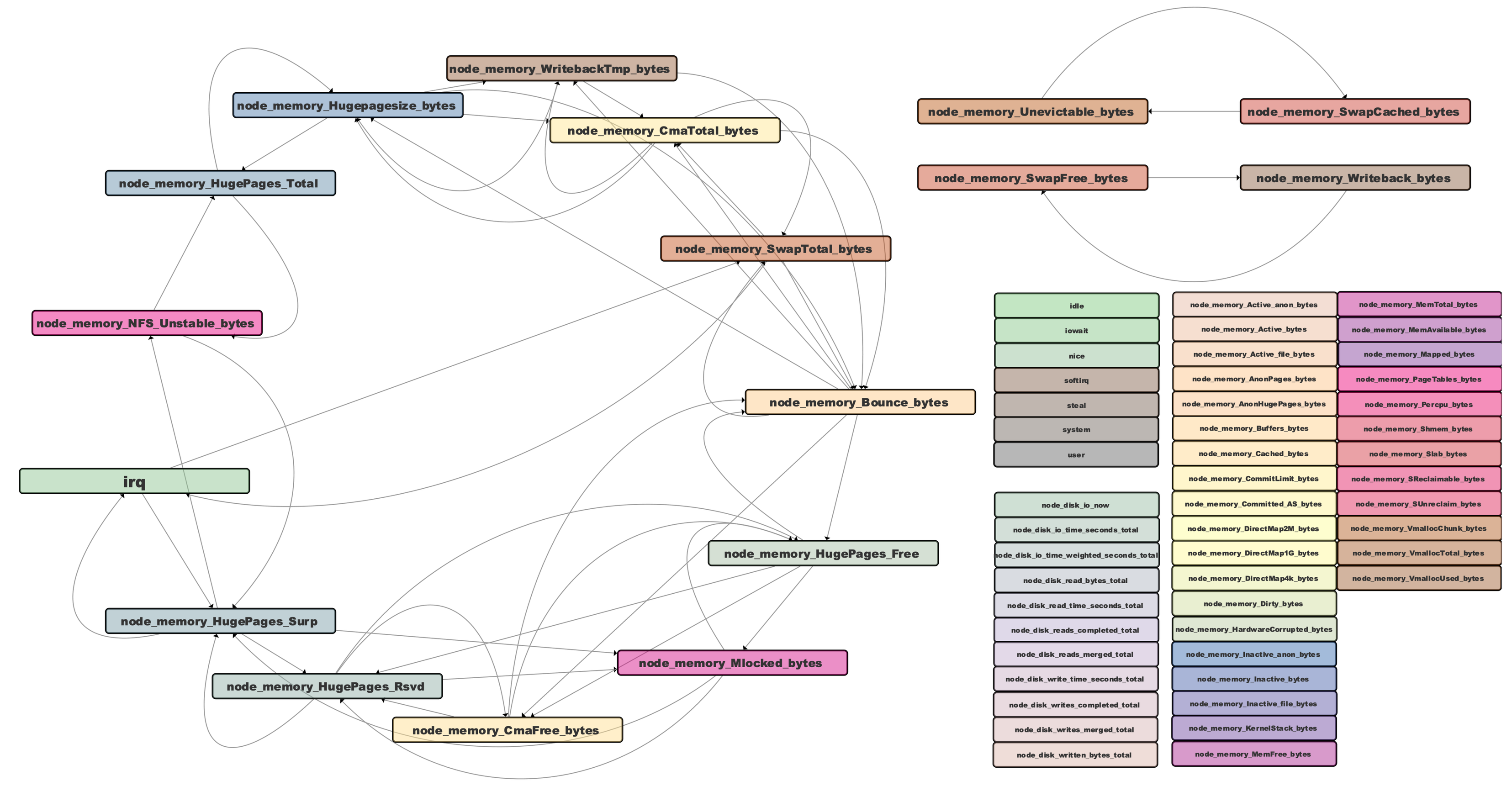
Evaluation

We investigate anomalies in a dataset collected from 332 compute nodes, comprising a total of 181GB of monitored metrics. All compute nodes can be categorized into five groups (i.e., G1-G5), each with distinct hardware characteristics.

❖ Learned graph relations

Directed graph construction based on all the monitored metrics within a compute node.

- prune the fully connected graph by considering the **cosine similarity** and the **number of neighbors** for each node within the graph.



Learned graph relations with 66 monitored metrics from CPU, memory and disk from all compute nodes in G1

❖ Efficiency in detecting synthetic anomalies

- Injecting noise with gaussian distribution on different monitored metrics to conduct a synthetic dataset with anomalies.
- The threshold of deviations (τ) is set as a specific centile of the normal data.
- Report *precision* and *F1 Score* for anomalous node identification and *accuracy of root cause* analysis, defined as the ratio of successful root cause identifications to the predicted anomalies.

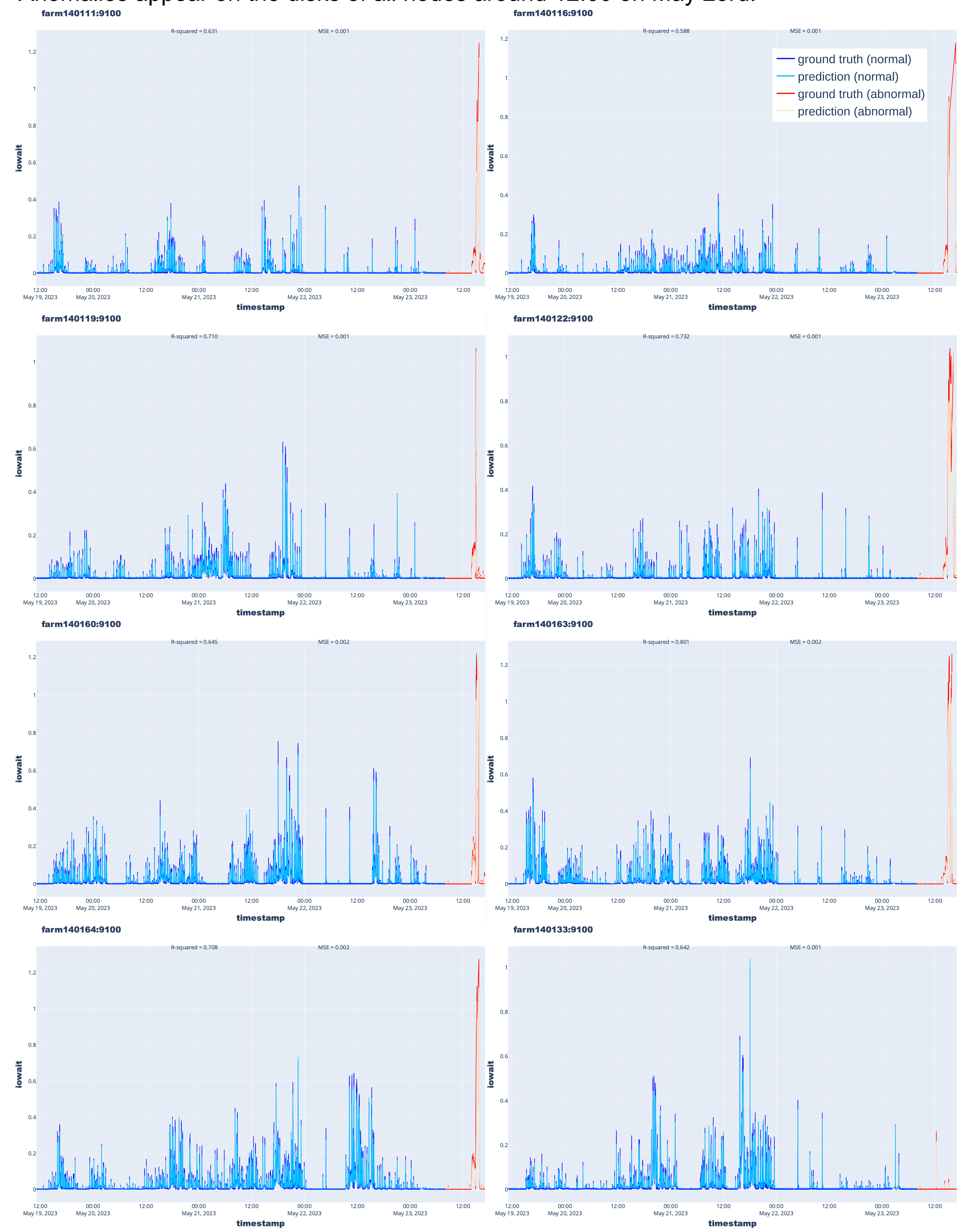
			G1	G2	G3	G4	G5	Average
CPU	Precision (F1 Score)	τ = p99.99	0.72 (0.8)	0.72 (0.68)	0.92 (0.73)	1 (0.95)	1 (1)	0.87 (0.83)
		τ = p100	1 (0.18)	1 (0.10)	1 (0.57)	1 (0.67)	1 (0.92)	1 (0.49)
	Root Cause Accuracy	τ = p99.99	0.68	0.56	0.92	1	1	0.83
		τ = p100	1	1	0.875	1	1	0.98
Memory	Precision (F1 Score)	τ = p99.99	0.74 (0.85)	0.86 (0.9)	1 (0.71)	1 (0.89)	1 (1)	0.92 (0.87)
		τ = p100	1 (0.86)	1 (0.1)	0 (0)	1 (0.57)	1 (0.92)	0.8 (0.49)
	Root Cause Accuracy	τ = p99.99	0.74	0.82	1	1	1	0.91
		τ = p100	1	1	0	1	1	0.8
Disk	Precision (F1 Score)	τ = p99.99	0.63 (0.62)	0.79 (0.86)	1 (0.92)	1 (0.92)	1 (1)	0.88 (0.86)
		τ = p100	0 (0)	1 (0.1)	0 (0)	1 (0.4)	1 (0.71)	0.6 (0.24)
	Root Cause Accuracy	τ = p99.99	0.47	0.75	1	1	1	0.84
		τ = p100	0	1	0	1	1	0.6

Observation:

- ✓ We successfully detected anomalies and their root causes in most cases.
- ✓ '0' value indicates there is no anomalies been detected due to the overconfident uncertainty estimate.

❖ Efficiency in detecting anomalies on real-world cluster

The plot illustrates the predictions versus true values of *iowait* for eight nodes over time. Anomalies appear on the disks of all nodes around 12:00 on May 23rd.



Observation :

- ✓ The GNN model's mean squared error (MSE) on real-world data is just 0.001.
- ✓ The model accurately detects anomalies, including nodes that lack a clearly anomalous signature such as farm140133, aligning with user-level abnormal event logging.

Conclusion and Future Work

- By learning complex dependencies between monitored metrics and adopting attention mechanisms, the GNN model accurately identifies anomalous behavior.
- Root cause analysis further allows for quickly pinpointing anomalous within a node.
- In future work, we aim to enable continual learning with GNN to detect anomalies with varying morphology in the complex, dynamic compute cluster.

[1] RUAD: unsupervised anomaly detection in HPC systems, Martin Molan, Andrea Borghesi, Daniele Cesarini, Luca Benini, Andrea Bartolini , arxiv,2023