A GENERAL ANOMALY DETECTION FRAMEWORK FOR FLEET-BASED CONDITION MONITORING OF MACHINES

Kilian Hendrickx, Wannes Meert, Yves Mollet, Johan Gyselinck, Bram Cornelis,
Konstantinos Gryllias, Jesse Davis

PROJECT FOCUS

EARLY FAULT DETECTION IN MACHINE FLEETS BEFORE FAILURE OCCURS

KEY INNOVATION

- FLEET-BASED APPROACH LEVERAGING SIMILAR MACHINES
- UNSUPERVISED ANOMALY DETECTION WITHOUT HISTORICAL LABELED DATA
- REAL-TIME ONLINE MONITORING WITH VISUALIZATION

PROBLEM UNDERSTANDING

Performance and reliability of machines are important in industrial situations. Faults and malfunctions may lead to production loss or even loss of life. This project focuses on finding the faulty machines before failure.

DRAWBACKS OF PRE-EXISTING SOLUTIONS:

1. Handcrafting indicators- Time consuming, prone to human errors, useless for complex systems
2. supervised learning - Require large historical labled dataset, still neededs human involvment
3. Semisupervised - Doesnt consider machine degradation, hard to determine enviranmental factors

NEED FOR EDGE?

Large Data Volume issues

- High frequency vibration signals
- Electrical measurements (continuous monitoring)
- Multiple sensor streams per machine

For human emergencies and monitoring,

- Real-time fault detection required
- Immediate response to prevent damage

Bandwidth limitations

- Local Processing: Reduced data transmission
- Low Latency: Immediate anomaly detection
- Low Latency: Immediate anomaly detection

OUR SOLUTION

4 step framework:
Raw Sensor Data → [1] Machine Comparison → [2] Fleet Clustering → [3] Anomaly Detection → [4] Visualization

Step 1: Machine Comparison

- Pairwise similarity measurement
- Domain-specific preprocessing
- Normalization techniques

Step 2: Fleet Clustering

- Hierarchical clustering
- Automatic cluster number determination
- Graph visualization for interpretability

This step essentially captures the similarity between machines using unsupervised learning to detect anomalies in machines based on the working conditions of other machines.

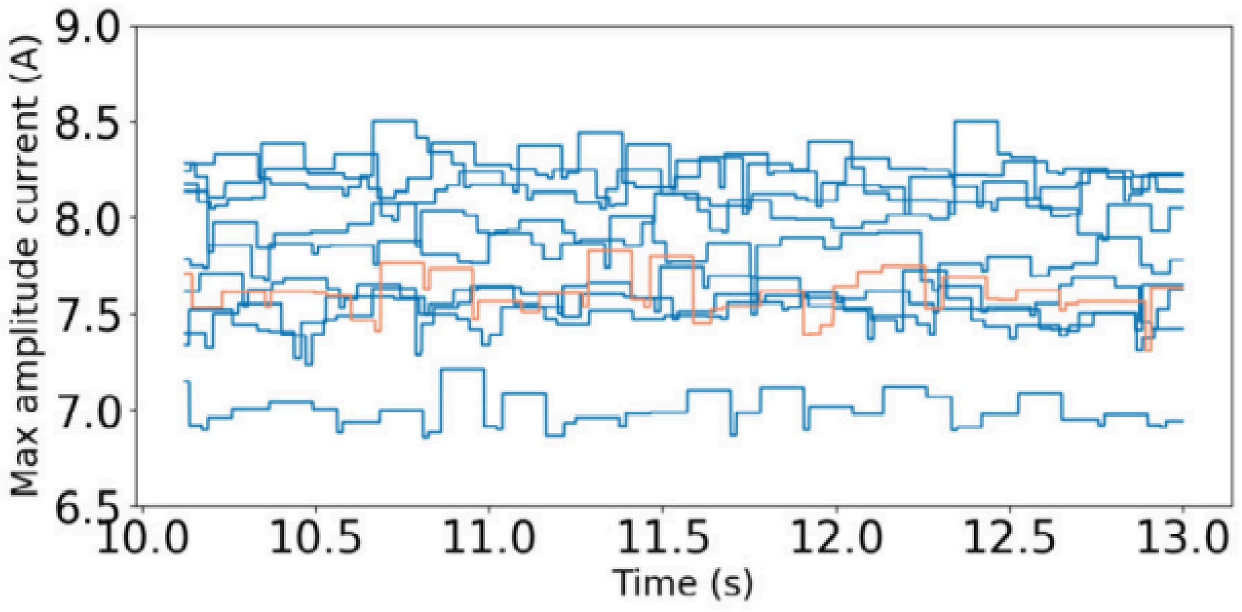
OUR SOLUTION

Step 3: Anomaly Detection

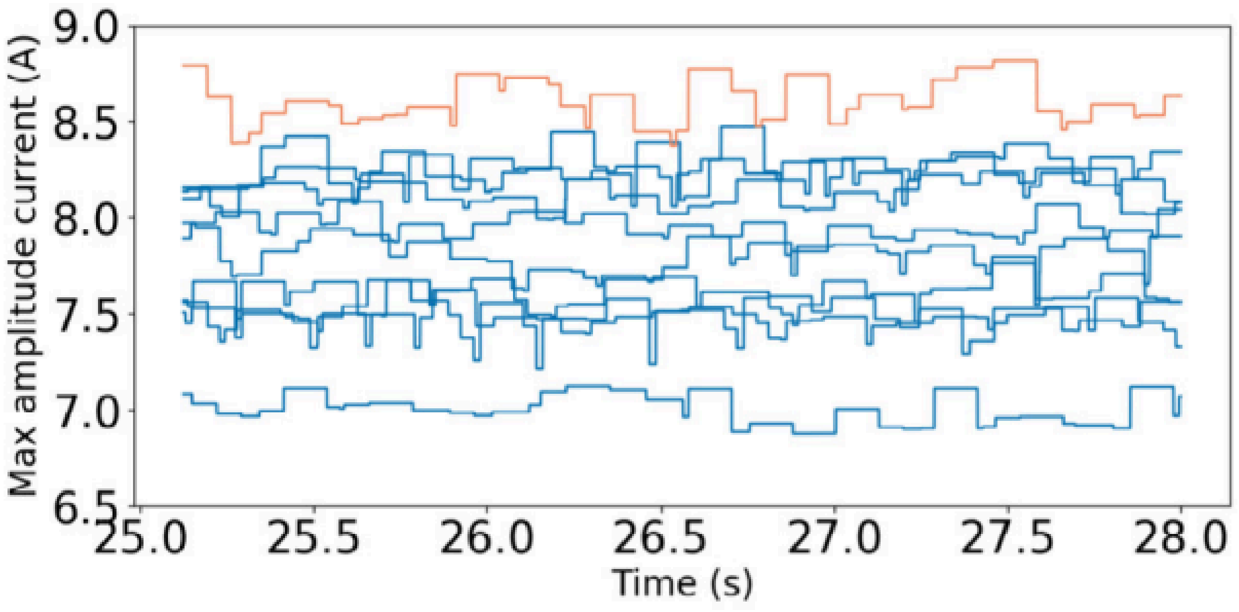
- Anomaly score = fraction of machines outside each cluster
- Assumption: majority of machines are healthy
- Threshold-based classification

Step 4: Visualization

- Interactive dashboards for domain experts
- Signal analysis, similarity matrices
- Interpretable results for validation

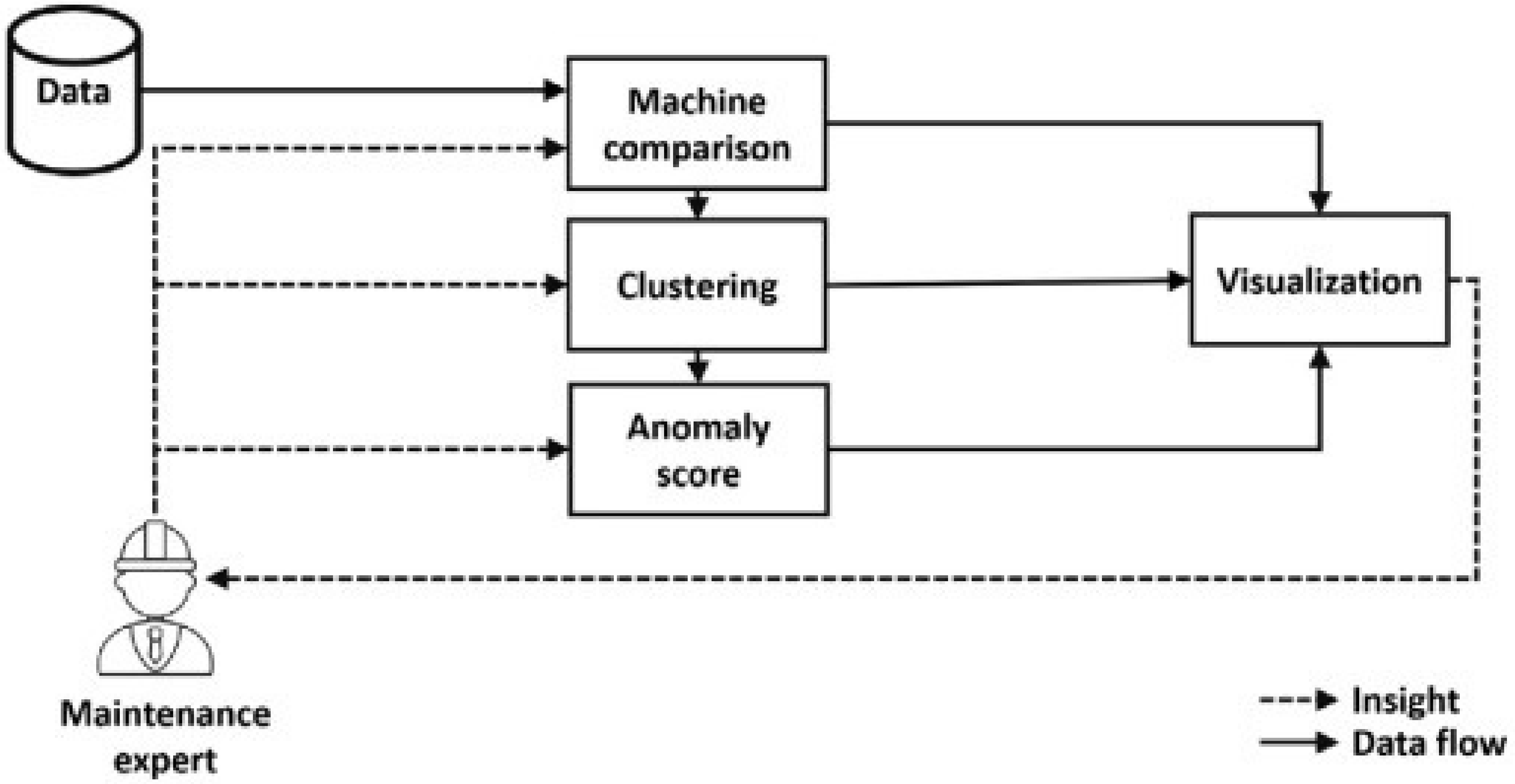


(a) All machines healthy

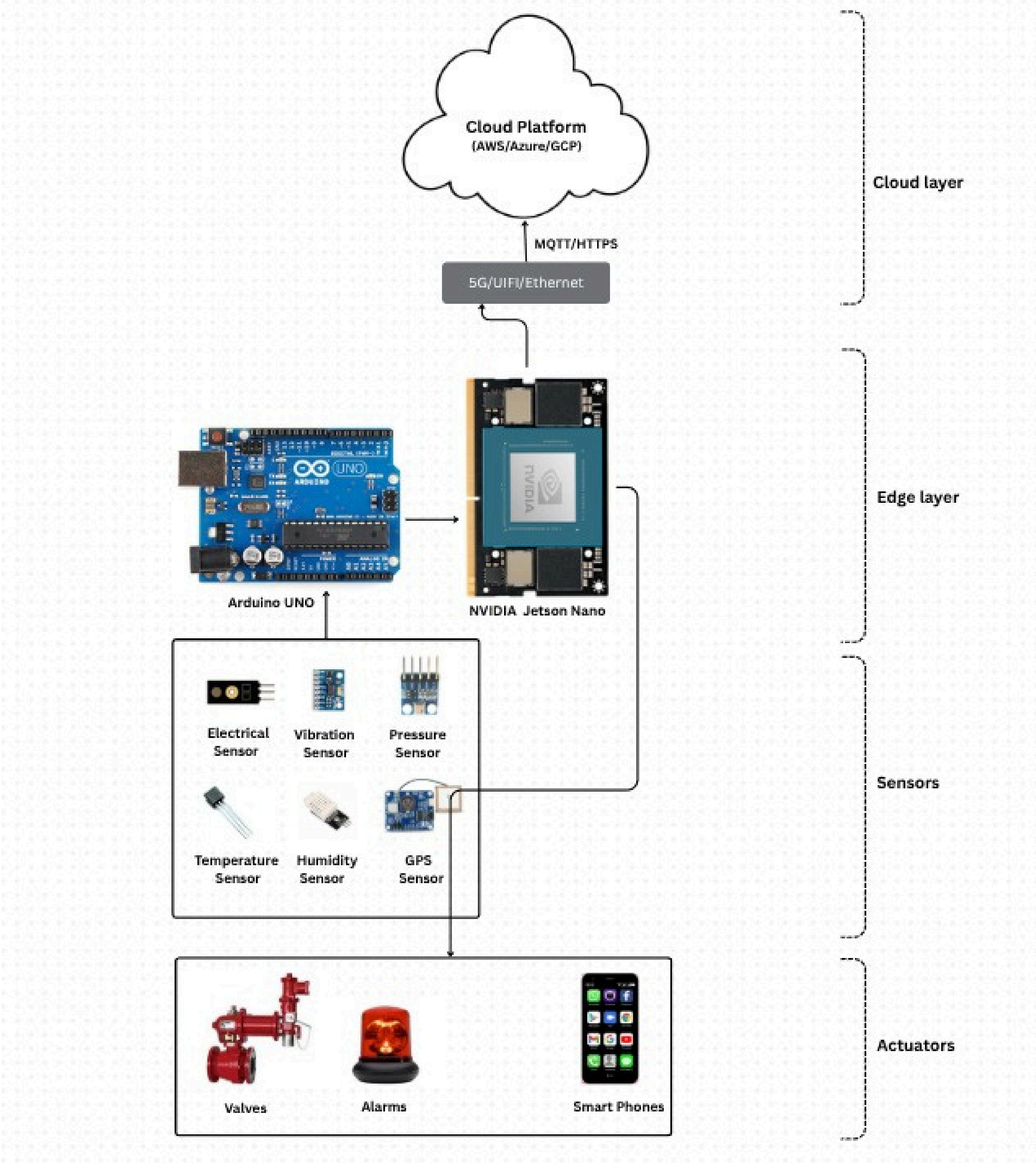


(b) D2_10 (orange) faulty

OUR SOLUTION



The expert is necessary to correctly calibrate the machines based on working conditions for the initial setup and to check if majority of the machines are working correctly and can be relied on to compare with other machines.



This architechture diagram demonstrates how this system works at various levels.

Clustering Methodology

1. Data Normalization

Min-Max Scaling: Adjusts signal amplitudes to a consistent range.

Percentile-Based Scaling: Enhances robustness against outliers.

Domain-Specific Preprocessing: Tailored to time, frequency, and time-frequency domains.

2. Similarity Measurement

Euclidean Distance: Applied in low-dimensional feature spaces.

Dynamic Time Warping (DTW): Captures time-series shape deviations.

Warping Amount Metric: Quantifies deviation in temporal alignment.

3. Hierarchical Clustering

Linkage Strategy: Single-linkage clustering to maintain tight groupings.

Cluster Validation: Cophenetic correlation coefficient optimizes cut-off thresholds.

Interpretability: Dendrogram visualizations aid expert assessment.

STATE OF THE ART & LITERATURE

Recent Advances in Fleet Monitoring

Traditional Methods:

- MCSA, vibration-based monitoring, statistical control

AI Approaches:

- Deep learning for fault detection (Zhao et al., 2019)
- SVMs for anomaly detection (Gryllias & Antoniadis, 2012)
- Semi-supervised structural health monitoring (Rogers et al., 2019)

Our Key Contributions

- First unsupervised fleet-level framework for electrical signature analysis
- Introduced DTW warping amount for shape deviation
- Cophenetic correlation for automatic clustering
- Multi-domain analysis: time, frequency, time-frequency
- Highly interpretable over black-box AI models

Validation Highlights

- Tested on a 10-machine electrical drivetrain fleet
- Detected voltage unbalance via electrical & vibration signatures
- Outperformed traditional MCSA methods

TASK ALLOCATION & NEXT STEPS

Sariga - Create dataset for different clusters, visualise and verify it belongs to the specified cluster and task scheduling.

Dwarakesh - Implement algorithm for preprocessing data, find model or create model for comparing 2 machines and clustering.

Shyam Sundar - Takes care of the simulation environment in Simpy, integrates the AI models and algorithms.

Pooja Shree - Implements heartbeat and sensor failure mechanism and load balancing.