Intelligent Asset Management: Predictive Maintenance of Heavy Mining Fleet via IoT and AI

L PRASHANTH KUMAR[1], P LOKESWARA RAO[2], B NAVAJEEVAN[3]
[1, 2,3,]Assistant Professor, Department of Mining Engineering, Anu Bose Institute of Technology for women's, KSP Road, New Paloncha, Bhadradri Kothagudem District, Telangana (TS), 507115

Abstract — Unexpected failures of haul trucks, conveyors, crushers, and excavators disrupt production, increase safety risks, and incur repair costs substantially higher than planned maintenance. Conversely, fixed-interval servicing can replace components before their useful life has ended. This paper describes the design and simulation-based evaluation of an Internet of Things (IoT) and machine-learning (ML) framework for predictive maintenance of mining equipment. The framework integrates sensing nodes for vibration, bearing temperature, oil pressure, oil temperature, motor current, and shaft speed; an edge gateway that extracts statistical and trend-based features; a hybrid wireless and wired network backbone; and cloud analytics that identify failures expected within 72 hours and estimate remaining useful life (RUL). The study develops analytical models for bearing life, Weibull reliability, health index, evaluation metrics, alarm logic, edge data reduction, availability, and maintenance cost. A run-to-failure dataset comprising 60 machines, 49,869 records, and 35 features was synthesised, and the models were validated using 18 machines excluded from training. Random Forest provided the strongest overall failure-classification results, achieving 81.8% precision, 86.7% recall, an F1-score of 0.842, and an AUC of 0.995. Gradient Boosting produced the most accurate RUL estimates, with an RMSE of 36.2 hours and an R^2 of 0.870. Using a two-out-of-three alarm rule, the system alerted operators before failure for all 18 test machines, with a mean lead time of 77.0 hours, a minimum of 42 hours, and 0.16 false-alarm episodes per 1,000 operating hours. Edge-based feature extraction reduced transmitted data volume by 99.998%. According to the cost model, predictive maintenance lowered total maintenance costs to US$0.24 million, compared with US$0.62 million for preventive maintenance and US$1.59 million for reactive maintenance, while increasing availability from 97.95% to 99.63%. These findings suggest that a well-engineered IoT-ML system can shift mining maintenance from a reactive approach to a predictive one.

Keywords: Predictive maintenance, mining equipment, Internet of Things, machine learning, remaining useful life, condition monitoring, edge computing, Random Forest

This is an open access article under the creative commons license https://creativecommons.org/licenses/by-nc-nd/4.0/
CC BY-NC-ND 4.0

I. INTRODUCTION

1.1 Background of the Mining Industry

Mining supplies the coal, metals and industrial minerals on which power generation, construction, manufacturing and transport depend. Modern mines are capital-intensive operations built around large fleets of mobile and fixed equipment: electric rope shovels, hydraulic excavators, rigid-frame haul trucks, drills, belt conveyors, crushers and grinding mills. Because ore bodies are increasingly lower in grade and deeper, this equipment must run for long hours at high utilisation in dusty, hot, wet and vibration-prone surroundings, often far from workshops and spare-part stores [1]. The availability of a few critical machines frequently decides the output of the entire mine.

1.2 Importance of Predictive Maintenance

Maintenance strategies are commonly grouped as reactive (run to failure), preventive (time- or usage-based), condition-based and predictive [1], [2]. Reactive maintenance is simple but exposes the mine to unplanned stoppages, secondary damage and safety incidents. Preventive maintenance reduces surprise failures but replaces parts according to a conservative calendar, so a large part of the remaining

component life is discarded. Predictive maintenance (PdM) uses measured condition data to estimate when a component will actually fail, so that repair can be scheduled just before failure at a convenient time, with parts and crew ready. In a mine this means fewer haul-road blockages, shorter stoppages of conveyor lines that feed the plant, longer component life and safer work, because inspection and repair are planned rather than emergency tasks.

1.3 Existing Challenges in Mining Operations

Conventional maintenance depends on periodic manual inspection, hand-held vibration meters, oil-sampling laboratories and the experience of fitters. These methods leave long intervals in which the machine is unobserved and are subjective. Applying PdM in a mine is also difficult for practical reasons: sites are remote with limited and intermittent connectivity; underground radio propagation is demanding and devices must satisfy intrinsic-safety limits [16], [17]; sensors are exposed to shock, dust and moisture; failure events are rare, so training data are highly imbalanced; and fleets are heterogeneous, so a model trained on one machine may not transfer to another. Streaming raw high-frequency vibration to a distant server is also costly in bandwidth and energy, which motivates computation at the edge [14].

1.4 Research Problem and Motivation

The Industrial IoT [13], [24] offers low-cost connected sensing, and ML offers data-driven diagnosis and prognosis [3]–[5]. However, a mine-grade PdM solution has to reconcile several demands at once: early enough warning to plan a repair, a low false-alarm rate so that maintenance crews trust the system, bounded data traffic over a constrained network, and a demonstrable economic benefit. This paper is motivated by the need for a quantified, design-level understanding of how these elements interact, from sensor to maintenance decision, for the type of equipment found in a mine.

1.5 Research Objectives and Contributions

The principal contributions of this work are:

- A layered IoT architecture linking equipment-mounted sensing nodes, an edge gateway, a hybrid network and a cloud analytics and maintenance-management layer.
- Analytical models for signal features, bearing life, Weibull reliability, health index, evaluation metrics, edge data reduction and maintenance cost.
- A two-stage ML scheme that combines failure-within-72-h classification and RUL regression, with a two-out-of-three alarm rule for false-alarm control.
- A comparative study of five classifiers and three RUL regressors using machine-grouped validation, so that results reflect unseen equipment.
- An economic and availability comparison of reactive, preventive and predictive strategies.

II. LITERATURE SURVEY

2.1 Review of Existing Research and Methodologies

Jardine et al. [2] reviewed machinery diagnostics and prognostics for condition-based maintenance and defined the data-acquisition, data-processing and decision-making stages that most later systems follow. Lei et al. [3] surveyed machinery health prognostics from data acquisition to RUL prediction and classified approaches as physics-based, statistical and AI-based. Lee et al. [23] described the prognostics and health management cycle for rotary machinery, and Randall and Antoni [22] provided a widely used tutorial on envelope and spectral analysis for rolling-element bearings, which remain the most common failure site in motors, gearboxes and conveyor idlers.

On the ML side, Carvalho et al. [4] systematically reviewed ML methods for predictive maintenance and found that Random Forest, support-vector machines and neural networks are the most frequently applied. Gebraeel et al. [7] estimated residual life from vibration degradation signals with neural networks. Zhao et al. [5] and Zhang et al. [6] reviewed deep-learning methods for machine health monitoring and bearing diagnostics, noting their need for large labelled datasets. Benchmark run-to-failure simulation work such as that of Saxena et al. [8] has shown how degradation can be modelled to evaluate prognostic algorithms when field data are scarce.

For connectivity, Xu et al. [13] reviewed IoT in industries, Shi et al. [14] outlined the case for edge computing to reduce latency and bandwidth, and MQTT [15] has become a standard lightweight publish-subscribe protocol for telemetry. In mining, Misra et al. [16] discussed safety and rescue-communication systems in high-stress environments, and Moridi et al. [17] compared ZigBee network topologies for underground monitoring, showing that topology strongly affects delivery performance.

2.2 Analysis of Existing Technologies and Comparison of Findings

Table 1: Comparison of representative previous studies

Study	Focus	Key contribution	Limitation for this work
Jardine et al. [2]	CBM diagnostics and prognostics review	Defined acquisition, processing and decision stages	General industry; no mining duty cycle
Lei et al. [3]	RUL prognostics review	Physics, statistical and AI taxonomy	Conceptual; no IoT or cost model
Carvalho et al. [4]	ML for PdM (literature review)	RF, SVM, ANN most used	Few harsh remote-site cases
Gebraeel et al. [7]	Neural RUL from vibration	Residual life from degradation signals	Bearing-level; no network layer
Zhao et al. [5]	Deep learning for health monitoring	Surveyed CNN, RNN, autoencoders	Needs large labelled data
Xu et al. [13]	IoT in industries	Architecture and enabling technologies	Limited analytics depth
Misra et al. [16]	Mining safety and communication	Resilience and post-event operation	Not equipment-health focused

2.3 Research Gaps and Limitations

- Most PdM studies address either the analytics or the connectivity, and few connect sensing, edge processing, network load and maintenance economics in one quantified model.
- Many studies report record-level accuracy only, whereas maintenance planners need alarm lead time and false alarms per operating hour.
- Validation frequently mixes records of the same machine between training and test sets, which overstates performance on unseen equipment.
- The cost and availability consequences of the prediction errors are rarely compared with preventive and reactive practice using the same data.

III. PROBLEM STATEMENT

The specific problem addressed is the design of a monitoring and analytics system that detects the onset of mechanical and thermal degradation in mining equipment early enough to schedule repair, delivers the information reliably over a constrained network, limits nuisance alarms, and demonstrates a lower cost of maintenance than reactive and calendar-based practice.

Table 2: Limitations of conventional maintenance and the required improvement

Aspect	Conventional practice	Required improvement
Trigger	Failure or fixed calendar interval	Condition and predicted failure time
Visibility	Periodic manual rounds	Continuous multi-sensor monitoring
Component life	Parts replaced early, life wasted	Replace near end of useful life
Downtime	Long, unplanned stoppages	Short, planned stoppages
Data	Paper logs, limited trending	Time-series store and ML analytics
Alarm quality	Single threshold, many false alarms	Feature fusion and confirmation rule

The technical challenges are (i) strong load- and environment-dependent variation of sensor signals, (ii) extreme class imbalance because failures are rare, (iii) generalisation to machines not seen during training, (iv) bandwidth and energy limits on the edge network, and (v) conversion of model output into a maintenance decision with measurable benefit. Operational challenges include sensor installation and calibration, integration with the computerised maintenance management system (CMMS), and acceptance by maintenance crews.

IV. OBJECTIVES OF THE STUDY

Primary objective: To design and evaluate an IoT and machine-learning framework for predictive maintenance of mining equipment.

Secondary objectives:

1. To identify the condition parameters, sensing techniques and alert levels for critical rotating mining equipment.
2. To model degradation, reliability and signal features analytically and by simulation.
3. To compare classical and neural ML models for failure prediction and RUL estimation on unseen machines.
4. To quantify alarm lead time, false-alarm rate and edge data reduction.
5. To compare reactive, preventive and predictive strategies in cost, downtime and availability.

Expected technical improvements: early warning measured in days rather than hours, a false-alarm rate low enough for crew acceptance, a data-traffic reduction of several orders of magnitude, and lower total maintenance cost. **Scope:** The study covers sensing, networking, ML analytics, alarm logic and economic evaluation on a synthetic run-to-failure fleet. Field trials, explosion-proof certification of hardware and deep sequence models are outside the present scope.

V. PROPOSED METHODOLOGY

5.1 Description of the Proposed Approach

The proposed approach is a hierarchical IoT pipeline. Sensing nodes on the machine measure vibration, temperature, pressure, current and speed. An edge gateway computes features, applies local thresholds

and forwards compact packets over a wireless mesh, private LTE or wired link to a cloud or control-room server using MQTT [15]. The server stores the data, runs the ML models for failure probability and RUL, applies the alarm rule and issues work orders to the CMMS. The workflow is shown in Fig. 1.

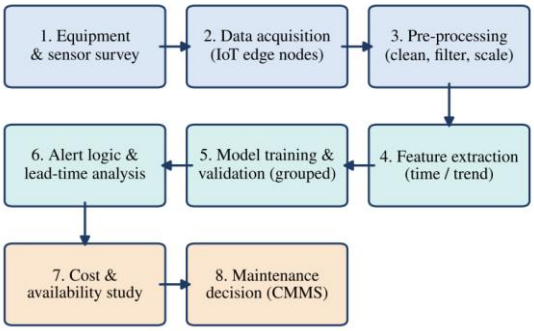


Fig. 1: Research methodology and workflow of the proposed predictive maintenance system

5.2 Selection of Sensing Techniques and Components

Sensor selection follows the dominant failure modes. Piezoelectric accelerometers capture bearing and gear vibration, from which RMS and kurtosis are derived; resistance temperature detectors (RTDs) monitor bearing and oil temperature; a pressure transducer monitors lubrication pressure; a current transformer measures motor load; and a tachometer measures shaft speed. Table 3 lists the monitored parameters and the illustrative alert levels. Actual limits must follow the equipment manufacturer and standards such as ISO 10816-3 [19] and ISO 13373-1 [20].

Table 3: Monitored parameters, sensing technique and illustrative alert levels

Parameter	Sensor	Range	Normal / Alert	Rate
Vibration RMS	Accelerometer	0–20 mm/s	≤ 4.5 / 7.1 mm/s	10 s
Vibration kurtosis	Accelerometer	1–15	≤ 3.5 / 5.0	10 s
Bearing temp.	RTD Pt100	0–150 °C	≤ 80 / 95 °C	30 s
Oil pressure	Pressure transducer	0–10 bar	≥ 3.8 / ≤ 3.2 bar	30 s
Oil temp.	RTD Pt100	0–150 °C	≤ 80 / 90 °C	30 s
Motor current	Current transformer	0–300 A	≤ 160 / 175 A	10 s
Shaft speed	Tachometer	0–3000 rpm	1480 ± 1%	10 s

5.3 Analytical and Simulation Procedure

The evaluation was carried out in five steps. (1) A fleet of 60 machines was simulated; each machine has a Weibull-distributed life, a random degradation onset between 45% and 70% of life, and a power-law degradation curve. (2) Sensor signals were generated as functions of load, ambient temperature, degradation and noise. (3) Rolling features were computed. (4) Classifiers and regressors were trained

on 42 machines and tested on 18 machines unseen in training (machine-grouped split). (5) The alarm rule, economic model and edge data-reduction model were evaluated on the test machines. The parameters are given in Table 4. All results are simulation outcomes, not field measurements.

Table 4: Simulation parameters

Parameter	Value
Machines (train / test)	60 (42 / 18)
Life distribution	Weibull, $\beta \approx 2.4$, scale 1,800 h, limited to 700–3,600 h
Degradation onset	45–70% of life; exponent 1.6–2.6
Sampling interval (model)	2 h (aggregated edge features)
Records / features	49,869 / 35
Label (classification)	Failure within 72 h (4.2% positive)
RUL target	Capped at 400 h
Rolling window	12 samples (24 h)
Alarm rule	2 of 3 consecutive, $\theta = 0.5$
Planned / unplanned repair cost	US$6,000 / US$45,000
Planned / unplanned downtime	6 h / 36 h
Downtime cost	US$1,200 per hour
False-alarm inspection	US$800 (2 h)
Preventive interval	0.7 × mean training life

5.4 Data Collection and Processing

Each node samples at the interval in Table 3. The gateway computes RMS, kurtosis and crest factor for the vibration channel and moving statistics for the others, then transmits a compact feature packet. At the server, values are checked against physical ranges, missing values are removed, and features are scaled using the training-set mean and standard deviation. For every sensor four rolling features are computed over a 12-sample window: mean, standard deviation, exponentially weighted moving average (EWMA) and linear slope. Together with the instantaneous value this gives five features per sensor and 35 in total.

5.5 Evaluation and Validation

Because records from one machine are strongly correlated, the data were split by machine, not by record. Performance is judged by accuracy, precision, recall, F1-score and AUC for classification (Eqs. 13–14), by RMSE, MAE and R^2 for RUL (Eq. 15), and by alarm lead time, false alarms per 1,000 hours, cost, downtime and availability (Eqs. 16–21). Field validation against a real fleet is identified as future work.

VI. SYSTEM ARCHITECTURE

Fig. 2 shows the overall architecture. The equipment layer contains the monitored assets. The edge layer comprises sensing nodes and a gateway that performs feature extraction and local alarms, so that a safety-relevant condition is flagged even if the uplink is lost. The network layer selects a link suited to each location: a wireless mesh on the haul road or in a plant, private LTE/5G in open pits, and fibre or

industrial Ethernet as the backbone and for underground trunk routes. The cloud or control-room layer hosts the MQTT broker, time-series database, ML engine, dashboards and CMMS interface.

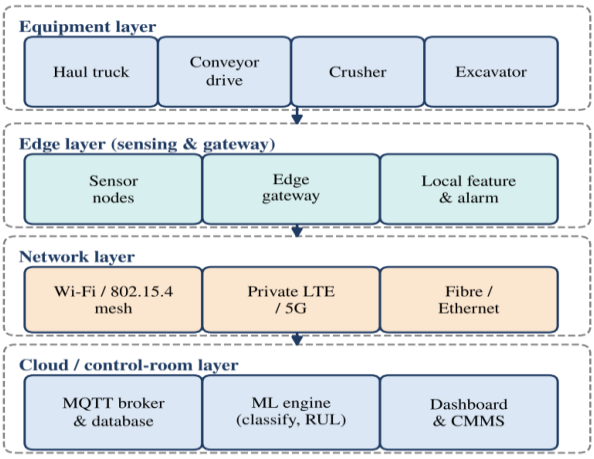


Fig. 2: Overall architecture of the IoT-based predictive maintenance system

A sensing node (Fig. 3) integrates the sensors, signal conditioning and an analogue-to-digital converter, a low-power microcontroller that computes features and thresholds, a radio, and a battery or energy-harvesting supply. Edge processing is essential because streaming raw vibration is impractical (Section 7.8).

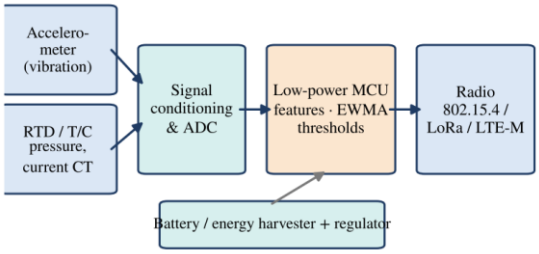


Fig. 3: Block diagram of a sensing node

Table 5: Typical characteristics of candidate communication technologies

Technology	Data rate	Range	Latency	Power
IEEE 802.15.4 mesh	250 kbit/s	10–100 m/hop	20–100 ms	Low
LoRaWAN	0.3–50 kbit/s	2–15 km (open)	1–2 s	Very low
Wi-Fi (802.11n/s)	50–600 Mbit/s	50–100 m	< 10 ms	High
Private LTE / 5G	10–100+ Mbit/s	0.5–5 km	10–50 ms	Medium
Fibre / Ethernet	0.1–10 Gbit/s	Km (wired)	< 1 ms	Wired

Table 5 shows why a hybrid network is used. Low-rate links such as LoRaWAN suffice for feature packets but not for waveform upload, whereas Wi-Fi and LTE can carry short waveform snapshots for expert diagnosis when an alarm occurs.

VII. MATHEMATICAL MODELLING AND TECHNICAL ANALYSIS

7.1 Vibration Features and Bearing Life

For a window of N vibration samples x_i with mean μ and standard deviation σ, the RMS value, kurtosis and crest factor are [22]

$$x_{\text{rms}} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2} \quad (1)$$

$$K = \frac{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^4}{\sigma^4} \quad (2)$$

$$CF = \frac{\max_i |x_i|}{x_{\text{rms}}} \quad (3)$$

RMS reflects overall vibration energy, kurtosis rises when impulsive bearing defects appear, and crest factor increases with isolated peaks. The basic rating life of a rolling bearing in hours is given by [21]

$$L_{10} = \left(\frac{C}{P}\right)^p \frac{10^6}{60\,n} \quad (4)$$

where C is the dynamic load rating, P the equivalent load, p the life exponent (3 for ball bearings) and n the speed in rpm. For C = 120 kN, P = 30 kN, p = 3 and n = 1,480 rpm, Eq. (4) gives $L_{10} = 4^3 \times 10^6 / (60 \times 1{,}480) \approx 721$ h, the time by which 10% of identical bearings are expected to fail under that heavy duty. Actual lives scatter widely, which motivates condition monitoring rather than reliance on the calculated value.

7.2 Pre-processing

Each sensor stream is smoothed with an exponentially weighted moving average and standardised using training statistics:

$$s_t = \lambda\, x_t + (1 - \lambda)\, s_{t-1} \quad (5)$$

$$z = \frac{x - \mu_{\text{train}}}{\sigma_{\text{train}}} \quad (6)$$

with $0 < \lambda \leq 1$ ($\lambda = 0.2$ here). Using training statistics only avoids information leakage from the test machines.

7.3 Reliability Model

Component life is modelled by the two-parameter Weibull distribution [18]. The reliability and hazard functions are

$$R(t) = \exp\left[-\left(\frac{t}{\eta}\right)^{\beta}\right] \quad (7)$$

$$h(t) = \frac{\beta}{\eta}\left(\frac{t}{\eta}\right)^{\beta - 1} \quad (8)$$

where β is the shape and η the scale parameter. Fitting the simulated lives of the 60 machines gives $\beta = 2.64$ and $\eta = 1{,}902$ h, so the mean life is $\eta\,\Gamma(1 + 1/\beta) \approx 1{,}690$ h. Because $\beta > 1$ the hazard increases with age (wear-out), which is why age-based replacement is wasteful for some units and late for others. For t = 1,000 h, R = 0.832 and $h = 0.48 \times 10^{-3}$ per h, whereas at t = 1,800 h, R = 0.421 and $h = 1.27 \times 10^{-3}$ per h (Fig. 4).

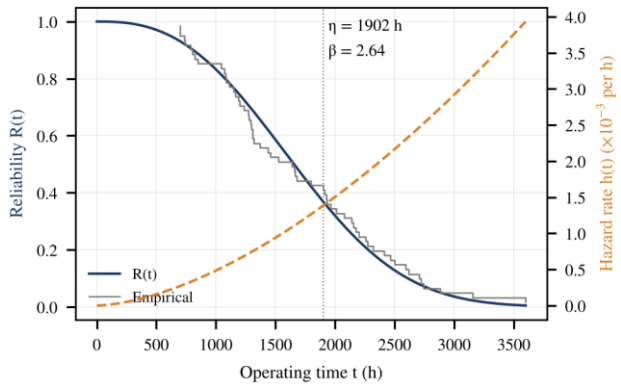


Fig. 4: Weibull reliability and hazard rate of the simulated fleet with empirical survival curve

7.4 Health Index and Remaining Useful Life

A normalised health index fuses m sensors between their safe and failure levels:

$$HI_t = \sum_{i=1}^{m} w_i \frac{x_{i,t} - x_i^{\text{safe}}}{x_i^{\text{fail}} - x_i^{\text{safe}}} \tag{9}$$

with weights w_i summing to one. If the failure time of a machine is T_f, its true RUL at time t is

$$RUL(t) = T_f - t \tag{10}$$

The RUL regressors are trained on the target of Eq. (10) capped at 400 h, because errors on machines that are still far from failure have little maintenance value.

7.5 Machine-Learning Models

Five classifiers were applied to the 35 features: logistic regression, a support-vector machine with RBF kernel, Random Forest [9], histogram-based Gradient Boosting [10] and a two-hidden-layer artificial neural network (64 and 32 neurons). Logistic regression estimates the failure probability as

$$P(y = 1 \mid \mathbf{x}) = \frac{1}{1 + e^{-(\mathbf{w}^T\mathbf{x} + b)}} \tag{11}$$

while each tree of the Random Forest chooses splits that minimise the Gini impurity

$$G = 1 - \sum_{k=1}^{K} p_k^2 \tag{12}$$

where p_k is the fraction of class k in a node. Class weights inversely proportional to class frequency were used to counter the imbalance (4.2% positive). Ridge regression, Random Forest and Gradient Boosting were used for RUL. Models were built with scikit-learn [12].

7.6 Performance Metrics

Precision, recall and F1-score are defined from true positives (TP), false positives (FP) and false negatives (FN):

$$P = \frac{TP}{TP + FP}, \quad R = \frac{TP}{TP + FN} \tag{13}$$

$$F_1 = \frac{2PR}{P + R} \tag{14}$$

For the RUL regression, the root-mean-square error between actual y_j and predicted $\hat{y}_j$ is

$$RMSE = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_j - \hat{y}_j)^2} \tag{15}$$

7.7 Alarm Logic and Lead Time

To suppress isolated spurious predictions, an alarm A_t is raised only when at least two of the last three predictions exceed the threshold θ:

$$A_t = 1 \text{ if } \sum_{k=0}^{2} [\hat{p}_{t-k} \geq \theta] \geq 2 \quad (16)$$

The warning lead time is the interval between the first alarm and the failure,

$$T_{\text{lead}} = T_f - T_{\text{alarm}} \quad (17)$$

and is the quantity that determines whether a repair can be planned. An evaluation window of 108 h (1.5 × the 72 h horizon) is used; alarms earlier than that window are counted as false-alarm episodes.

7.8 Edge Data Reduction and Latency

For a vibration channel sampled at f_s with b bits per sample and n_{ch} channels, the raw bit rate B_{raw} is compared with the bit rate of the feature packets B_{feat}, and the data-reduction ratio is

$$B_{\text{raw}} = f_s \, b \, n_{ch}, \quad \rho = 1 - \frac{B_{\text{feat}}}{B_{\text{raw}}} \quad (18)$$

With f_s = 25.6 kHz, b = 16 bit and one channel, B_{raw} = 409.6 kbit/s (about 4.42 GB per day). A feature packet of 64 bytes every 60 s needs only 8.53 bit/s (92 kB per day), so ρ = 99.998%. The end-to-end latency from measurement to decision is

$$T_{e2e} = T_{\text{sense}} + T_{\text{edge}} + T_{\text{tx}} + T_{\text{cloud}} + T_{\text{act}} \quad (19)$$

For assumed values of 1.0 s feature window, 0.05 s edge processing, 0.15 s private-LTE transmission, 0.30 s cloud inference and 0.5 s action, $T_{e2e} \approx 2.0$ s; with LoRaWAN (1.5 s transmission) it is about 3.4 s. Both are negligible against a warning lead time of days.

7.9 Availability and Maintenance Cost

Availability is the ratio of mean time between failures (MTBF) to the sum of MTBF and mean time to repair (MTTR),

$$A = \frac{MTBF}{MTBF + MTTR} \quad (20)$$

and the total maintenance cost of a strategy over the test fleet is

$$C = N_f(C_f + D_f c_d) + N_p(C_p + D_p c_d) + N_{fa} C_{fa} \quad (21)$$

where N_f, N_p and N_{fa} are the numbers of unplanned failures, planned interventions and false-alarm inspections, C_f, C_p and C_{fa} the corresponding direct costs, D_f and D_p the downtime hours and c_d the downtime cost per hour. With the Table 4 values, one unplanned failure costs 45,000 + 36 × 1,200 = US$88,200, one planned intervention costs 6,000 + 6 × 1,200 = US$13,200 and one false alarm costs US$800.

VIII. RESULTS AND DISCUSSION

Note on results. The results below were obtained from the simulated fleet and the assumed parameters of Table 4. They characterise design trade-offs and are not field measurements.

8.1 Sensor Behaviour During Degradation

Fig. 5 shows the signals of one simulated machine over its life. Vibration RMS and bearing temperature remain flat for roughly the first 60% of life and then rise with increasing slope, oil pressure falls, and motor current rises more slowly and with a large load-induced scatter. The last 72 h before failure (shaded) show the largest deviations, but they are only a small part of the record, which is why class imbalance is a central issue.

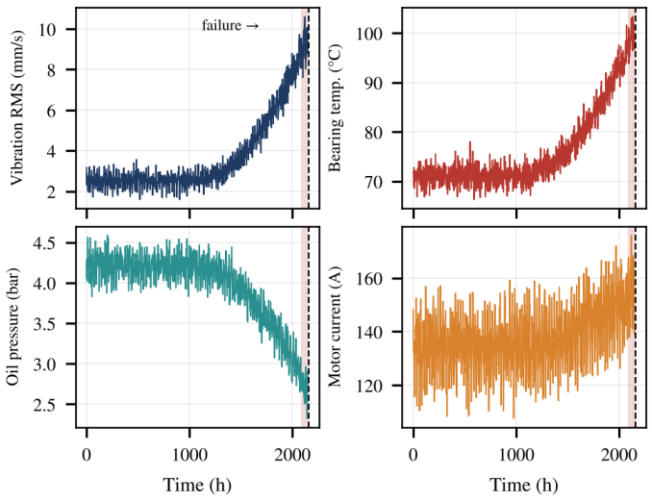


Fig. 5: Sensor signals of one simulated machine; the dashed line marks failure and the shaded band the last 72 h

8.2 Failure-Prediction Performance

Table 6: Classification performance on the 18 unseen test machines (15,609 records, 633 positive)

Model	Acc.	Prec.	Rec.	F1	AUC
Logistic Regression	0.971	0.583	0.970	0.728	0.996
SVM (RBF)	0.971	0.588	0.968	0.732	0.994
Random Forest	**0.987**	**0.818**	0.867	**0.842**	0.995
Gradient Boosting	0.982	0.709	0.932	0.805	0.996
ANN (MLP)	0.986	0.800	0.861	0.830	0.996

All models separate the classes well in terms of AUC (0.994–0.996, Fig. 6), so AUC alone does not discriminate between them. The differences appear at the operating threshold of 0.5 (Table 6, Fig. 8). Logistic regression and SVM reach a recall near 97% but a precision of only 58–59%, i.e. roughly four false alarms for every six true detections. Random Forest trades about ten points of recall for 23 points of precision and obtains the highest F1-score (0.842); the ANN is a close second (0.830). Gradient Boosting has the second-highest recall (93.2%) with precision of 70.9%, so it is attractive where missed failures are costlier than inspections.

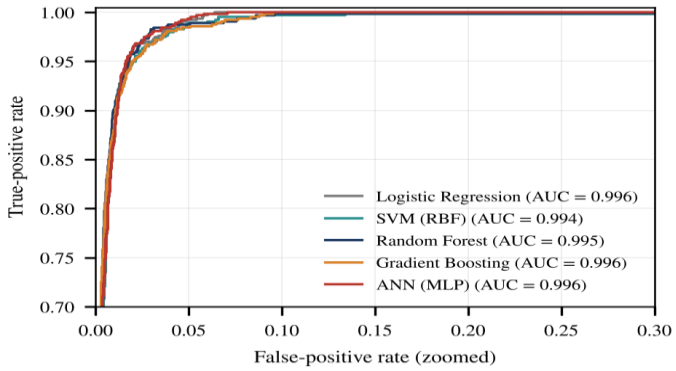


Fig. 6: ROC curves of the five classifiers (zoomed to the upper-left region)

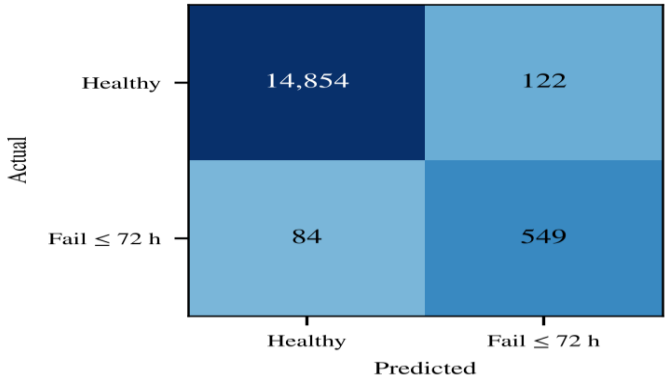


Fig. 7: Confusion matrix of the Random Forest classifier on the test machines

The Random Forest confusion matrix (Fig. 7) contains 549 true positives, 84 false negatives, 122 false positives and 14,854 true negatives, giving a record-level specificity of 99.19% and a false-positive rate of 0.81%. All 122 false positives lie within 157 h of failure, and 94 of them (77%) fall between 72 and 108 h before failure, just outside the labelled window, where the machine is already degraded and an early alert is harmless.

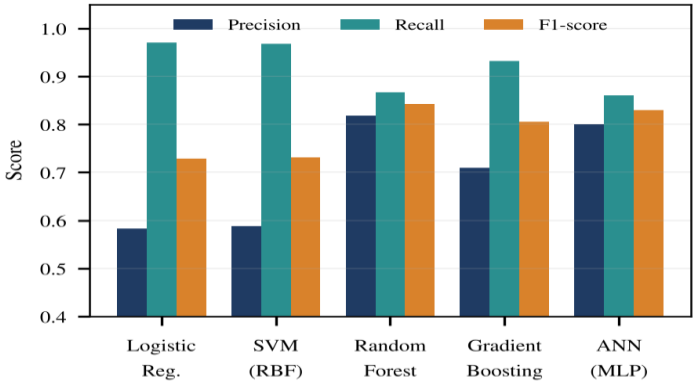


Fig. 8: Precision, recall and F1-score of the five classifiers

8.3 Feature Importance

Aggregating the Random Forest feature importances by sensor (Fig. 9) shows that vibration kurtosis contributes 27.9%, vibration RMS 18.1%, bearing temperature 17.3%, oil temperature 16.9% and oil pressure 12.4%, while shaft speed (4.4%) and motor current (2.9%) contribute little. The most important single features are kurtosis, its EWMA, the mean oil temperature and the EWMA of vibration RMS. Smoothed and trend features therefore carry more information than instantaneous readings, which supports the use of edge-side feature extraction. In the simulation, motor current and speed are dominated by load, so they are weak failure indicators; on real machines their relevance depends on the failure mode.

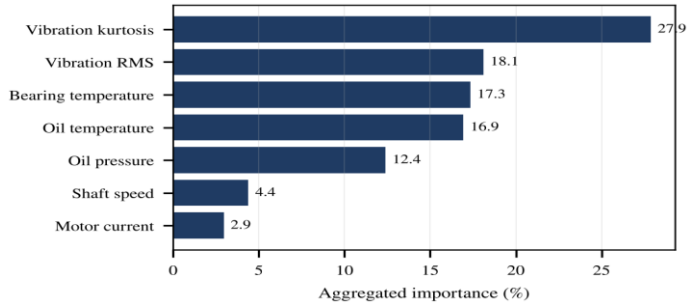


Fig. 9: Feature importance of the Random Forest model aggregated by sensor

8.4 Remaining Useful Life Estimation

Table 7: RUL regression performance on test machines (RUL capped at 400 h)

Model	RMSE (h)	MAE (h)	R^2	RMSE, RUL ≤ 150 h
Ridge Regression	39.1	24.5	0.848	55.2
Random Forest	37.5	18.5	0.860	53.4
Gradient Boosting	**36.2**	**17.9**	**0.870**	**49.0**

Gradient Boosting gave the lowest errors (RMSE 36.2 h, MAE 17.9 h, $R^2 = 0.870$; Table 7). The error rises to 49.0 h when only records within 150 h of failure are considered, because degradation accelerates and noise increases near failure. Fig. 10 shows that predictions follow the ideal line for RUL below about 250 h and spread at long RUL, where the signals are nearly flat and the models cannot distinguish a machine with 300 h from one with 400 h; this has little practical consequence because repair is scheduled only when RUL is short.

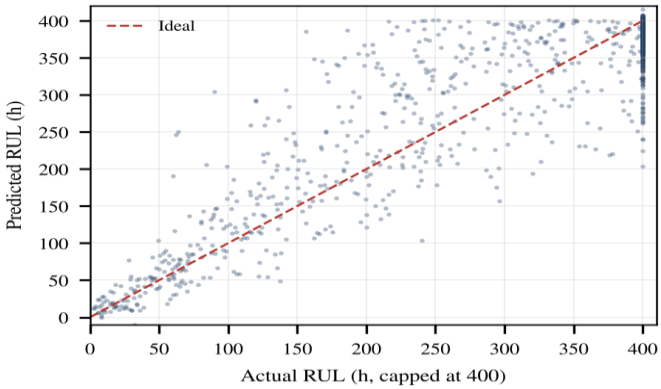


Fig. 10: Predicted versus actual RUL for the Gradient Boosting model

8.5 Alarm Lead Time and False Alarms

Applying Eq. (16) with the Random Forest probabilities, all 18 test machines were warned before failure. Fig. 11 illustrates one machine: the failure probability stays near zero for most of the life, begins to fluctuate about 150 h before failure, and the two-out-of-three rule fires 60 h before failure. Across the fleet (Fig. 12) the mean lead time was 77.0 h, the median 69.4 h and the minimum 42.0 h, which is sufficient to order parts and schedule a shift-change repair. Five false-alarm episodes occurred in 31,628 operating hours, i.e. 0.16 per 1,000 h, or roughly one unnecessary inspection per 6,300 fleet operating hours.

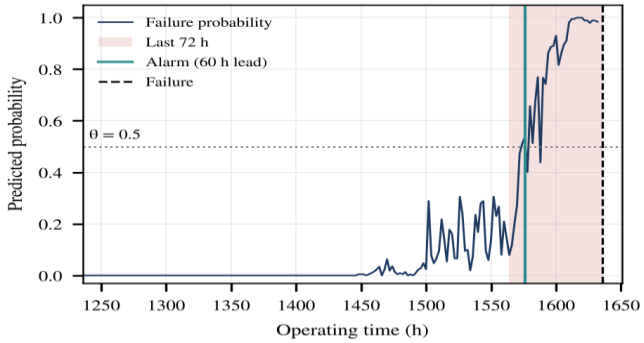


Fig. 11: Failure probability, alarm instant and failure for one test machine

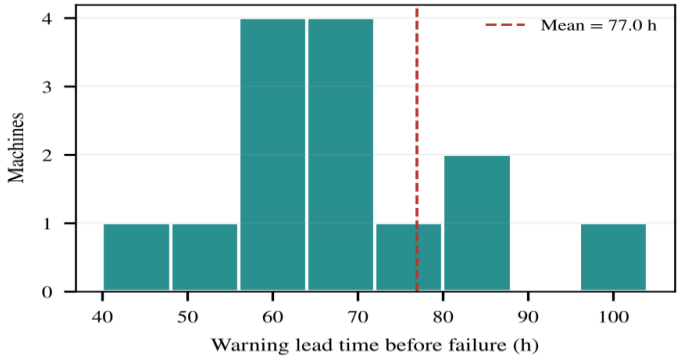


Fig. 12: Distribution of warning lead time over the 18 test machines

Table 8: Alarm performance summary (Random Forest, 2-of-3 rule)

Metric	Value
Machines warned before failure	18 of 18
Mean / median lead time	77.0 h / 69.4 h
Minimum lead time	42.0 h
False-alarm episodes	5 in 31,628 h
False-alarm rate	0.16 per 1,000 h
Record-level false-positive rate	0.81%

8.6 Cost and Availability Comparison

Table 9: Maintenance cost, downtime and availability of three strategies (18 machines, 31,628 h)

Strategy	Unplanned failures	Planned jobs	Cost (US$)	Downtime (h)	Avail. (%)
Reactive	18	0	1,587,600	648	97.95
Preventive (1,158 h)	4	20	616,800	264	99.17
Predictive	**0**	**18**	**241,600**	**118**	**99.63**

In reactive maintenance every machine fails once, costing US$1.59 million and 648 h of downtime. Preventive replacement every 1,158 h (0.7 of the mean training life) avoids most failures but still allows four early failures and performs 20 replacements, giving US$0.62 million. The predictive strategy intervened once per machine on warning, added five false-alarm inspections, and cost US$0.24 million, which is 84.8% below reactive and 60.8% below preventive maintenance. Downtime falls by 81.8% and 55.3% respectively, and availability rises to 99.63% (Table 9, Fig. 13). Savings depend on the assumed cost ratio of unplanned to planned work (88,200 to 13,200, about 6.7:1) and on the miss rate; with more missed failures the benefit shrinks, so the sensitivity of the model to this ratio must be checked with a mine's own data.

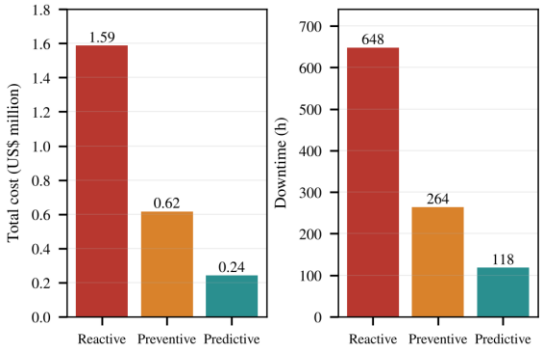


Fig. 13: Total maintenance cost and downtime of reactive, preventive and predictive strategies

8.7 Comparison with Existing Methods

Table 10: Summary comparison of maintenance approaches

Feature	Reactive	Preventive	Proposed PdM
Trigger	Failure	Calendar / hours	Predicted condition
Warning before failure	None	None (schedule)	77.0 h mean
Component life used	100%, with damage	About 70%	Close to full
Unplanned failures	18	4	0
Relative cost	100%	39%	15%
Data and tools	None	Records	IoT, edge, ML, CMMS
Main limitation	Downtime, safety	Wasted life	Data quality, model upkeep

The comparison shows that the principal benefits of the IoT-ML framework are earlier and plannable intervention, fuller use of component life, and lower cost, while its limitations are dependence on sensor quality, the need for failure-labelled history and periodic model retraining. These observations agree qualitatively with the PdM literature [3], [4]. As the evidence is simulated, absolute figures such as lead time and savings must be re-estimated for each mine, but the ranking of strategies and the importance of smoothed vibration and temperature features are expected to carry over.

IX. CONCLUSION

This paper presented the architecture, analytical models and simulation-based evaluation of an IoT and machine-learning framework for predictive maintenance of mining equipment. The key findings are as follows.

- Random Forest gave the best failure-prediction balance on unseen machines (precision 81.8%, recall 86.7%, F1 0.842), while Gradient Boosting estimated RUL with RMSE 36.2 h and $R^2 = 0.870$.
- A two-out-of-three alarm rule warned all 18 test machines with a mean lead time of 77.0 h and only 0.16 false-alarm episodes per 1,000 h.

- Smoothed vibration-kurtosis, vibration-RMS and temperature features were the strongest indicators, and edge extraction reduced data volume by 99.998%.
- Predictive maintenance reduced total cost to US$0.24 million versus US$0.62 million (preventive) and US$1.59 million (reactive), and raised availability to 99.63%.

The objectives of Section IV were thereby achieved at the simulation level. Practically, the framework supports parts ordering and repair scheduling days ahead, fewer emergency stoppages and safer maintenance.

Future Scope

- Validation on field data from operating haul trucks, conveyors and crushers with calibrated reference instruments.
- Deep sequence models such as LSTM networks [11] and transfer learning between machines of the same family.
- Explosion-proof and intrinsic-safety certification of node hardware for underground use.
- Integration with automatic work-order generation, spare-parts inventory and digital-twin visualisation.
- Uncertainty-aware RUL estimation and cost-sensitive thresholds optimised per machine.

REFERENCES

[1] R. K. Mobley, *An Introduction to Predictive Maintenance*, 2nd ed. Amsterdam, The Netherlands: Butterworth-Heinemann, 2002.
[2] A. K. S. Jardine, D. Lin, and D. Banjevic, "A review on machinery diagnostics and prognostics implementing condition-based maintenance," *Mech. Syst. Signal Process.*, vol. 20, no. 7, pp. 1483–1510, Oct. 2006.
[3] Y. Lei, N. Li, L. Guo, N. Li, T. Yan, and J. Lin, "Machinery health prognostics: A systematic review from data acquisition to RUL prediction," *Mech. Syst. Signal Process.*, vol. 104, pp. 799–834, May 2018.
[4] T. P. Carvalho, F. A. A. M. N. Soares, R. Vita, R. P. Francisco, J. P. Basto, and S. G. S. Alcalá, "A systematic literature review of machine learning methods applied to predictive maintenance," *Comput. Ind. Eng.*, vol. 137, Art. no. 106024, Nov. 2019.
[5] R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang, and R. X. Gao, "Deep learning and its applications to machine health monitoring," *Mech. Syst. Signal Process.*, vol. 115, pp. 213–237, Jan. 2019.
[6] S. Zhang, S. Zhang, B. Wang, and T. G. Habetler, "Deep learning algorithms for bearing fault diagnostics—A comprehensive review," *IEEE Access*, vol. 8, pp. 29857–29881, 2020.
[7] N. Gebraeel, M. Lawley, R. Liu, and V. Parmeshwaran, "Residual life predictions from vibration-based degradation signals: A neural network approach," *IEEE Trans. Ind. Electron.*, vol. 51, no. 3, pp. 694–700, Jun. 2004.
[8] A. Saxena, K. Goebel, D. Simon, and N. Eklund, "Damage propagation modeling for aircraft engine run-to-failure simulation," in *Proc. Int. Conf. Prognostics Health Manage.*, Denver, CO, USA, 2008, pp. 1–9.
[9] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, Oct. 2001.
[10] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Statist.*, vol. 29, no. 5, pp. 1189–1232, Oct. 2001.
[11] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.
[12] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011.
[13] L. Da Xu, W. He, and S. Li, "Internet of Things in industries: A survey," *IEEE Trans. Ind. Informat.*, vol. 10, no. 4, pp. 2233–2243, Nov. 2014.
[14] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, Oct. 2016.
[15] A. Banks, E. Briggs, K. Borgendale, and R. Gupta, Eds., *MQTT Version 5.0*, OASIS Standard, Mar. 2019.
[16] P. Misra, S. Kanhere, D. Ostry, and S. Jha, "Safety assurance and rescue communication systems in high-stress environments: A mining context," *IEEE Commun. Mag.*, vol. 48, no. 4, pp. 66–73, Apr. 2010.
[17] M. A. Moridi *et al.*, "Performance analysis of ZigBee network topologies for underground space monitoring and communication systems," *Tunn. Undergr. Space Technol.*, vol. 27, no. 1, pp. 69–75, 2012.

[18] W. Weibull, "A statistical distribution function of wide applicability," *J. Appl. Mech.*, vol. 18, no. 3, pp. 293–297, 1951.

[19] *Mechanical Vibration—Evaluation of Machine Vibration by Measurements on Non-Rotating Parts—Part 3*, ISO 10816-3, 2009.

[20] *Condition Monitoring and Diagnostics of Machines—Vibration Condition Monitoring—Part 1: General Procedures*, ISO 13373-1, 2002.

[21] *Rolling Bearings—Dynamic Load Ratings and Rating Life*, ISO 281, 2007.

[22] R. B. Randall and J. Antoni, "Rolling element bearing diagnostics—A tutorial," *Mech. Syst. Signal Process.*, vol. 25, no. 2, pp. 485–520, Feb. 2011.

[23] J. Lee, F. Wu, W. Zhao, M. Ghaffari, L. Liao, and D. Siegel, "Prognostics and health management design for rotary machinery systems—Reviews, methodology and applications," *Mech. Syst. Signal Process.*, vol. 42, no. 1–2, pp. 314–334, Jan. 2014.

[24] J. Lee, B. Bagheri, and H.-A. Kao, "A cyber-physical systems architecture for Industry 4.0-based manufacturing systems," *Manuf. Lett.*, vol. 3, pp. 18–23, Jan. 2015.