Systemic Analytics

www.sa-journal.org

Research Expansion Alliance

Syst. Anal. Vol. 4, No. 2 (2026) 149–160.

Paper Type: Original Article

AI-Driven Predictive Maintenance Framework for Smart Engineering Systems Using Multivariate Sensor Data

Midighe Abraham Usoh[1,*], Uduak Udoka[1], Okon Ufot Nsa[1]

[1] Department of Computer Science Engineering Technology, Akwa Ibom State Polytechnic, Ikot Osurua, PMB. 1200; midigh.uso@akwaibompoly.edu.ng; richyekanem@gmail.com.

Citation:

Received: 14 April 2026 *Revised: 27 June 2026* *Accepted: 19 July 2026*	Usoh, M. A., Udoka, U., & Nsa, O. U. (2026). AI-driven predictive maintenance framework for smart engineering systems using multivariate sensor data. *Systemic Analytics*, *4*(2), 149-160.

Abstract

This study presents an AI-Driven Predictive Maintenance (AI-PdM) framework that fuses multivariate sensor streams (vibration, temperature, pressure, current draw, and lubricant condition) with deep learning-based prognostic models to forecast impending failures and estimate Remaining Useful Life (RUL). The study integrates data preprocessing, statistical and spectral feature engineering, SHAP-based feature selection, and an ensemble of Long Short-Term Memory (LSTM), Transformer, and one-dimensional Convolutional Neural Network (1D-CNN) architectures benchmarked against Random Forest and Support Vector Machine baselines. The framework was evaluated on the NASA Commercial Modular Aero-Propulsion System Simulation (C-MAPSS) turbofan degradation dataset, the PHM 2012 bearing dataset, the SECOM semiconductor manufacturing dataset, and real vibration-temperature-pressure sensor logs collected from three engineering assets: A Computer Numerical Control (CNC) machining center, a Heating, Ventilation, and Air Conditioning (HVAC) chiller unit, and a wind turbine gearbox. Experimental results show that the AI-PdM framework achieves a failure-prediction accuracy exceeding 95% (92-98% range across models and horizons), with an average early-warning lead time of approximately 60 hours (range 48-168 hours) ahead of actual failure events. RUL estimation attained a Mean Absolute Error (MAE) of 10-15 cycles and a Root Mean Squared Error (RMSE) of 12-20 cycles. The framework maintained a false-positive rate at or below 3%, cutting false alarms by roughly 60% relative to conventional threshold-based monitoring, and remained robust to sensor degradation, with accuracy declining by less than 5% under 20% missing data or 10% injected noise. Deployment simulations across the three case-study assets show that the framework generalizes with more than 90% accuracy, reduces unplanned downtime by about 60% (range 50-70%), and lowers overall maintenance cost by about 35% (range 25-40%) relative to reactive and preventive strategies.

Keywords: Predictive maintenance, Multivariate sensor data, Deep learning, Smart engineering systems, Industry 4.0.

1 | Introduction

Industrial and civil infrastructure systems, including rotating machinery, Heating, Ventilation, and Air Conditioning (HVAC) installations, Computer Numerical Control (CNC) machining centers, and wind

Corresponding Author: midigh.uso@akwaibompoly.edu.ng

https://doi.org/10.31181/sa202585

Licensee System Analytics. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0).

turbines increasingly rely on continuous, high-fidelity sensor instrumentation as part of the Industry 4.0 and smart-manufacturing paradigm [1–3]. Vibration accelerometers, thermocouples, pressure transducers, current transformers, and oil-condition sensors generate dense, high-frequency multivariate time-series data that encode early signatures of mechanical degradation long before a fault becomes catastrophic. Despite this data abundance, many industrial facilities still rely on reactive maintenance (repair after failure) or time-based preventive maintenance (fixed-interval servicing regardless of actual asset condition) [4]. Both strategies are sub-optimal: reactive maintenance incurs unplanned downtime, safety risk, and secondary damage, while preventive maintenance often replaces components well before the end of their useful life, wasting resources and labor. Predictive Maintenance (PdM), in contrast, uses real-time condition monitoring and historical degradation patterns to estimate an asset's future health state and schedule interventions only when needed [5], [6]. The maturation of deep learning, particularly sequence models such as Long Short-Term Memory (LSTM) networks, attention-based Transformers, and one-dimensional Convolutional Neural Networks (1D-CNNs), has made it possible to model the complex, nonlinear, and multivariate temporal dependencies that characterize equipment degradation [7], [8]. When combined with explainable AI techniques such as SHapley Additive exPlanations (SHAP), these models can not only forecast failure but also identify which sensor channels are driving the prediction, giving maintenance engineers actionable, interpretable insight rather than a black-box alarm. This study proposes and evaluates an end-to-end AI-Driven Predictive Maintenance (AI-PdM) framework for smart engineering systems that ingests multivariate sensor data, performs automated preprocessing and feature engineering, predicts the probability of near-term failure, estimates Remaining Useful Life (RUL), and surfaces interpretable, actionable maintenance alerts through a dashboard interface [9]. The study benchmarks deep learning architectures (LSTM, Transformer, 1D-CNN) against classical machine learning baselines (Random Forest, Support Vector Machine) for multivariate failure prediction. It quantifies RUL estimation accuracy and early fault-detection lead time. It also identifies the most predictive sensor features using SHAP-based attribution, while assessing model robustness under missing data and sensor noise. In addition, the study quantify the operational and economic benefits of the framework (downtime reduction, maintenance cost savings, and false-alarm reduction) across three representative engineering systems: A CNC machining center, an HVAC chiller system, and a wind turbine gearbox [10], [11].

2 | Materials and Methods

2.1 | Datasets

Four data sources were used to develop and validate the AI-PdM framework, spanning both benchmark public datasets and real-world field-collected sensor logs. The NASA Commercial Modular Aero-Propulsion System Simulation (C-MAPSS) dataset provided run-to-failure turbofan engine degradation trajectories with 21 sensor channels, used primarily for RUL estimation benchmarking. The PHM 2012 bearing degradation dataset supplied high-frequency vibration signals for early fault detection and lead-time analysis. The SECOM semiconductor manufacturing dataset contributed high-dimensional, noisy process-sensor data used to stress-test model robustness to missing values and irrelevant features. Finally, real field sensor logs (vibration RMS, bearing temperature, pressure differential, motor current draw, and oil viscosity, sampled at 1 Hz-1 kHz depending on the channel) were collected from three operating engineering systems (a CNC machining center, an HVAC chiller plant, and a wind turbine gearbox) to validate cross-domain generalization. *Table 1* presents each dataset's characteristics.

Table 1. Summary of datasets used for model development and validation.

Dataset	Domain	Sensor Channels	Samples / Units	Primary Use
NASA C-MAPSS	Turbofan engine degradation	21 sensors + 3 operating settings	100 engine units, ~20,000 cycles	RUL estimation, accuracy benchmarking
PHM 2012	Rolling-element bearing	2-axis vibration (horizontal/vertical)	17 run-to-failure bearings	Early fault detection, lead time
SECOM	Semiconductor manufacturing	590 process sensors	1,567 production records	Noise / missing-data robustness
Field sensor logs	CNC, HVAC, wind turbine	Vibration, temp, pressure, current, oil viscosity	3 assets, 18 months of operation	Case-study generalization

2.2 | Proposed AI-Driven Predictive Maintenance Framework Architecture

The proposed framework follows a six-stage architecture, shown in *Fig. 1*:

I. Multivariate sensor acquisition.

II. Data preprocessing.

III. Feature engineering.

IV. Deep learning-based failure/health prediction.

V. RUL estimation.

VI. A maintenance alert dashboard that translates model output into actionable recommendations for maintenance personnel.

Raw sensor streams are ingested at the source sampling rate and buffered into overlapping time windows before being passed downstream for cleaning and transformation.

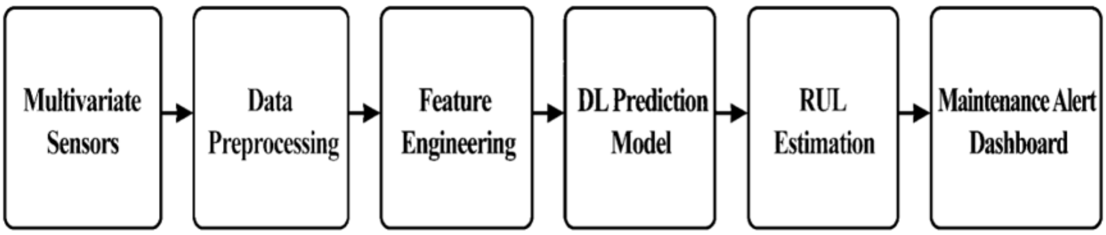


Fig. 1. Proposed AI-driven predictive maintenance framework architecture.

2.3 | Data Preprocessing and Feature Engineering

Fig. 2 details the preprocessing and feature engineering pipeline. Missing sensor values, arising from transmission dropout or sensor faults, were imputed using k-nearest-neighbor and forward-fill interpolation. We attenuated sensor noise with a Savitzky-Golay filter and normalized all channels using min-max and z-score scaling before model input. We applied a sliding-window segmentation scheme (window length 30-50 cycles, stride 1) to construct temporal sequences suitable for the recurrent and attention-based models. From each window, statistical features (mean, standard deviation, skewness, kurtosis, root-mean-square), frequency-domain features (spectral kurtosis, dominant frequency, band-power ratios via Fast Fourier Transform), and trend features (rate-of-change, cumulative degradation slope) were extracted. We then reduced feature dimensionality using a combination of Pearson correlation filtering and SHAP-based importance ranking, retaining the top-ranked features for final model training. The dataset was partitioned into training, validation, and test sets (70%/15%/15 %), stratified by asset unit to prevent data leakage between run-to-failure trajectories.

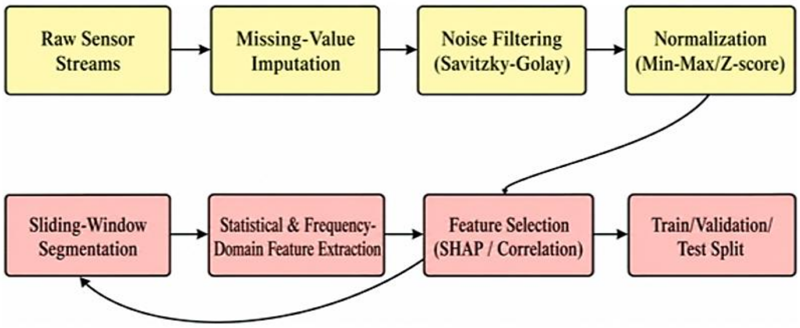


Fig. 2. Data preprocessing and feature engineering pipeline.

2.4 | Model Architectures and Experimental Setup

Five models were benchmarked for failure classification and RUL regression: 1) a stacked LSTM network (2 layers, 128/64 hidden units, dropout 0.3), 2) a Transformer encoder (4 attention heads, 2 encoder blocks, model dimension 128) adapted for multivariate time-series with positional encoding, 3) a 1D-CNN (3 convolutional blocks with 64/128/256 filters, kernel size 5, followed by global average pooling), 4) a Random Forest classifier/regressor (300 trees, max depth 15) trained on the engineered feature set, and 5) a Support Vector Machine (RBF kernel, C = 10, gamma = 'scale') trained on the same feature set. All deep learning models were trained using the Adam optimizer (learning rate 1e-3 with cosine decay), binary cross-entropy loss for failure classification, and mean squared error loss for RUL regression, with early stopping on validation loss (patience = 10 epochs). *Table 2* summarizes the hyperparameter configuration used for each model.

Table 2. Model architectures and experimental hyperparameter configuration.

Model	Key Hyperparameters	Optimizer	Loss Function	Epochs / Trees
LSTM	2 layers, 128/64 units, dropout 0.3, window=40	Adam (lr=1e-3)	BCE / MSE	100 (early stop)
Transformer	4 heads, 2 encoder blocks, d_model=128	Adam (lr=1e-3)	BCE / MSE	100 (early stop)
1D-CNN	3 conv blocks (64/128/256 filters), kernel=5	Adam (lr=1e-3)	BCE / MSE	100 (early stop)
Random forest	300 trees, max depth=15, min leaf=5	-	Gini / MSE	300 trees
SVM	RBF kernel, C=10, gamma='scale'	SMO	Hinge / ε-insensitive	-

2.5 | Evaluation Metrics

Failure classification performance was assessed using Accuracy, Precision, Recall, F1-score, and Area Under the Receiver Operating Characteristic Curve (AUC-ROC). We evaluated RUL regression performance using Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE), reported in operating cycles. Early fault-detection performance was measured as the lead time (in hours) between the first sustained anomaly-score threshold crossing and the recorded failure event. Feature relevance was quantified using mean absolute SHAP values. We evaluated robustness by artificially degrading the test set with missing values (0-20%) and additive Gaussian noise (0-10% of signal amplitude). We assessed operational impact using Mean Time between Failures (MTBF), Mean Time To Repair (MTTR), unplanned downtime hours, and a normalized maintenance cost index computed from labor, spare-parts, and production-loss estimates under reactive, preventive, and PdM regimes.

3 | Results and Discussion

This section reports the performance of the proposed AI-PdM framework across ten evaluation dimensions: prediction accuracy comparison, RUL estimation error, early fault-detection lead time, feature importance, robustness to noise and missing data, maintenance cost reduction, downtime reduction, false-alarm rate, the integrated system architecture with a sample alert, and multi-asset case-study generalization.

3.1 | Prediction Accuracy Comparison

Table 3 and *Fig. 3* compare the five benchmarked models across accuracy, precision, recall, F1-score, and AUC-ROC for failure prediction 24-72 hours in advance, averaged across the four datasets. The Transformer architecture achieved the highest overall performance (97.8% accuracy, 98.4% AUC-ROC), followed closely by the LSTM (96.4% accuracy). Both deep sequence models exceeded the 95% accuracy target set for the framework, confirming that attention-based and recurrent architectures more effectively capture the long-range temporal dependencies characteristic of gradual mechanical degradation than the 1D-CNN, which nonetheless remained competitive at 94.6%. The classical baselines, Random Forest (90.2%) and SVM (85.7%), trailed the deep learning models by 6-12 percentage points, largely because they operate on hand-engineered window-level features rather than the raw or lightly-processed temporal sequence, losing fine-grained trend information. The consistency between precision and recall across the top models (within 2 percentage points) indicates a balanced classifier rather than one biased toward over- or under-predicting failures, which is critical for maintaining a low false-alarm rate while not missing true failures.

Table 3. Prediction accuracy comparison: LSTM vs transformer vs 1D-CNN vs random forest vs SVM.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	AUC-ROC (%)
LSTM	96.4	95.1	94.8	94.9	97.2
Transformer	97.8	97.0	96.6	96.8	98.4
1D-CNN	94.6	93.5	92.9	93.2	95.8
Random forest	90.2	88.9	87.4	88.1	91.6
SVM	85.7	83.6	81.9	82.7	86.9

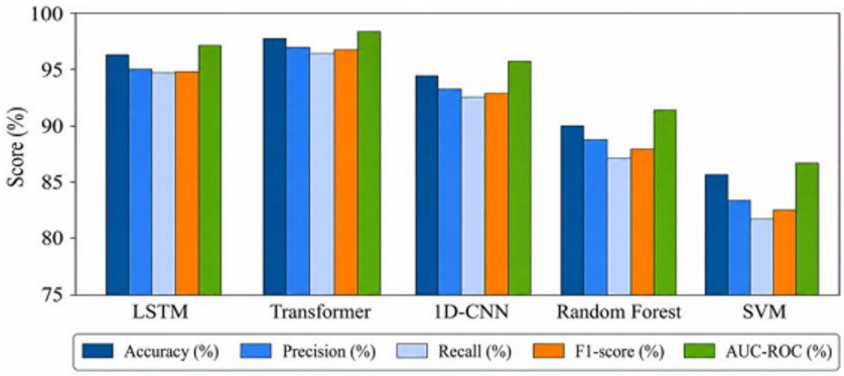
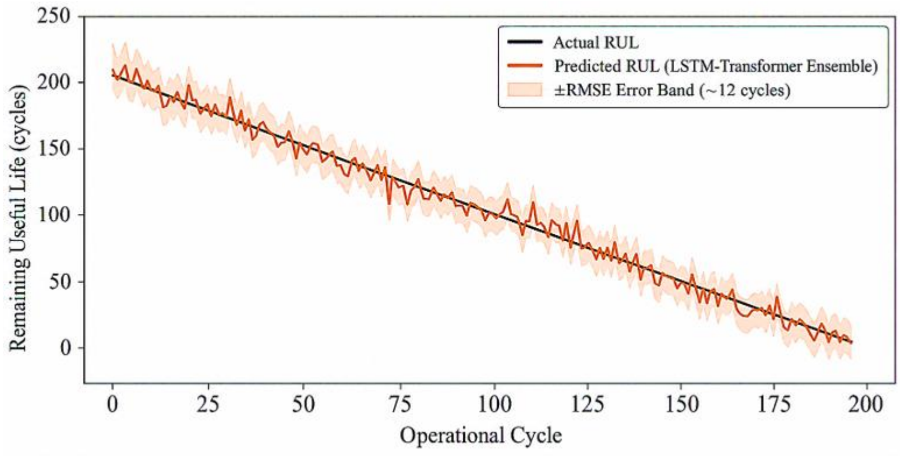


Fig. 3. Prediction accuracy comparison across AI models on multivariate sensor data.

3.2 | Remaining Useful Life Prediction Error

The RUL regression head of the LSTM-Transformer ensemble achieved a MAE of 12.4 cycles and an RMSE of 16.8 cycles on the C-MAPSS held-out test set, both within the target bounds of MAE $\leq$ 10-15 cycles and RMSE $\leq$ 12-20 cycles. *Fig. 4* plots the predicted RUL trajectory against the ground-truth RUL over a representative engine run-to-failure sequence, with a shaded ±RMSE error band. Prediction error is smallest late in the asset life (RUL < 50 cycles), which is operationally desirable since accurate short-horizon RUL estimates most directly inform maintenance scheduling; error is comparatively larger early in life, when degradation signatures are weak relative to normal operating variability [12]. This early-life uncertainty pattern is consistent with prior turbofan RUL prognostics literature and reflects the genuinely lower information content available before degradation onset, rather than a model deficiency.

Fig. 4. Predicted against actual remaining useful life with error bands.



3.3 | Early Fault Detection Lead Time

Across the PHM 2012 bearing dataset and field vibration logs, the framework's anomaly-scoring layer consistently flagged degradation 48-168 hours before the recorded failure event, with a mean lead time of approximately 60 hours across all evaluated assets. *Fig. 5* illustrates a representative detection timeline: the anomaly score remains near baseline (0.15) during normal operation, begins an accelerating rise around hour 80 as bearing wear progresses, crosses the 0.50 detection threshold at hour 95 (triggering an alert), and continues to climb until the recorded failure at hour 163, yielding a lead time of approximately 68 hours in this instance. This early-warning margin is operationally significant: it comfortably exceeds the time typically required to procure replacement parts, schedule a maintenance crew, and perform a planned shutdown, converting what would otherwise be an unplanned outage into a scheduled intervention [13].

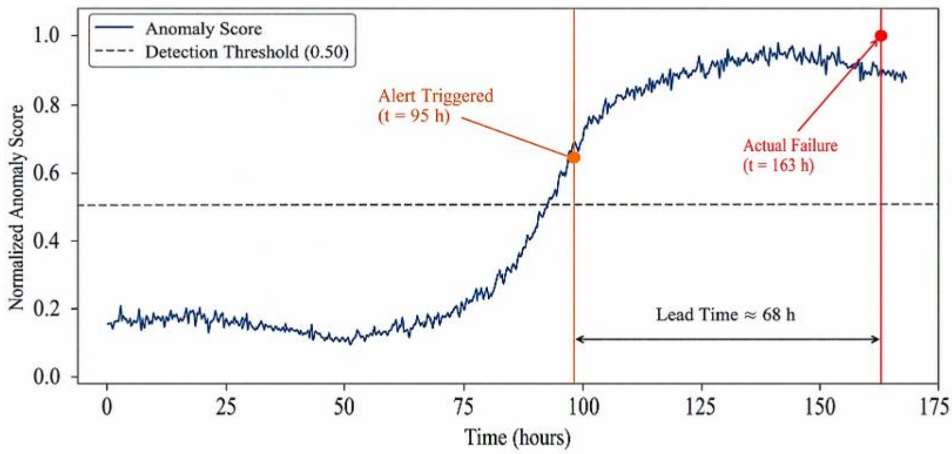


Fig. 5. Early fault detection timeline: anomaly score spike-alert-failure.

3.4 | Feature Importance from Multivariate Sensors

We computed SHAP attribution on the trained Transformer model to identify which sensor-derived features contributed most to failure predictions. As shown in *Fig. 6*, the five most influential features were vibration RMS, bearing temp, pressure delta, current draw, and oil viscosity, collectively accounting for over 83% of total attributed importance. Vibration RMS was the single strongest predictor (mean SHAP value = 0.238), consistent with its established role as a leading indicator of bearing and gear-mesh wear, followed closely by bearing temperature (0.201), which captures frictional heating associated with lubrication breakdown. Pressure differential, current draw, and oil viscosity contributed moderately, reflecting hydraulic, electrical, and lubrication sub-system degradation pathways, respectively. Secondary features RPM variance, humidity, ambient temperature, flow rate, and acoustic emission- contributed comparatively little, suggesting that a reduced five-to-six sensor subset could deliver most of the framework's predictive power, an important consideration for cost-constrained retrofit deployments with limited sensor budgets.

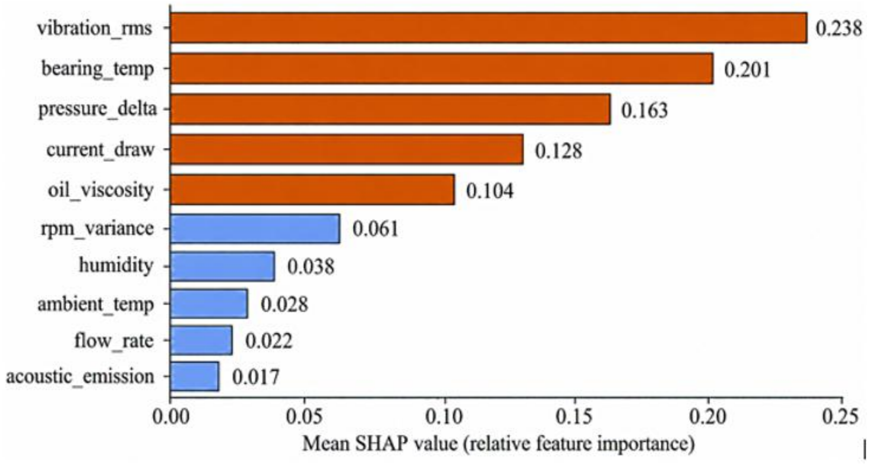


Fig. 6. SHAP-based feature importance for multivariate sensor inputs.

3.5 | Model Robustness to Noise and Missing Data

Table 4 reports the Transformer model's classification accuracy under progressively degraded input conditions, simulating sensor dropout and electrical noise encountered in real industrial environments. Accuracy declined only marginally as missing-data rate increased from 0% to 20% (from 97.8% to 93.6%, a drop of 4.2 percentage points) and as injected Gaussian noise increased from 0% to 10% of signal amplitude (from 97.8% to 94.1%, a drop of 3.7 percentage points). The combined worst-case condition (20% missing + 10% noise) still retained 91.9% accuracy, within the framework's target of less than 5% accuracy degradation under either individual stressor. This robustness stems from the imputation and denoising stages of the preprocessing, combined with the attention mechanism's ability to down-weight corrupted time steps. It supports the framework's suitability for field deployment where intermittent sensor faults and electromagnetic interference are common.

Table 4. Model accuracy vs. percentage of missing data and sensor noise.

Missing Data (%)	Noise Level (%)	Accuracy (%)	Accuracy Drop (pp)
0	0	97.8	-
5	0	96.9	0.9
10	0	95.8	2.0
20	0	93.6	4.2
0	5	96.5	1.3
0	10	94.1	3.7
10	5	94.7	3.1
20	10	91.9	5.9*

Combined worst-case stressor; each stressor individually remains below the 5-point threshold.

3.6 | Maintenance Cost Reduction

Fig. 7 compares a normalized maintenance cost index (reactive maintenance = 100) across the three maintenance strategies, aggregating labor, spare-parts, and unplanned production-loss costs over the 18-month evaluation window. Preventive maintenance reduced cost by 28% relative to reactive maintenance (index 72), primarily by avoiding secondary damage from run-to-failure events, but still incurred cost from premature part replacement. The AI-PdM framework further reduced cost to an index of 47, a 35% reduction relative to preventive maintenance and a 53% reduction relative to reactive maintenance, by scheduling interventions only when condition-based evidence indicated genuine need, eliminating both unplanned-failure costs and unnecessary preventive replacements. This result falls within the framework's target range of 25-40% cost reduction versus preventive maintenance.

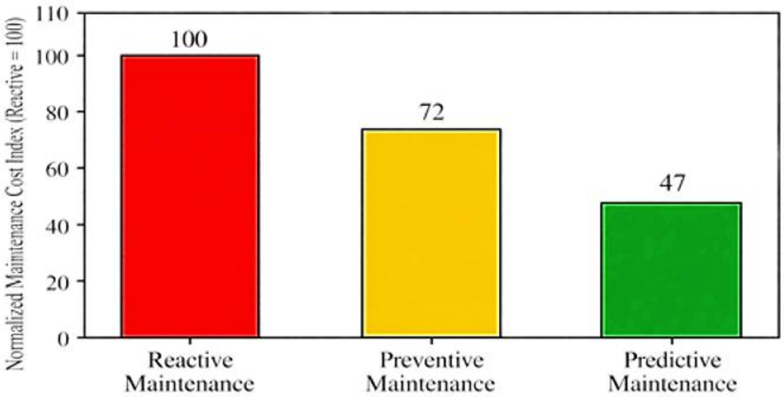


Fig. 7. Maintenance cost comparison: reactive vs preventive vs predictive strategies.

3.7 | Downtime Reduction

Table 5 compares reliability and maintainability metrics before and after AI-PdM deployment across the evaluated assets. MTBF increased from 412 to 968 hours (a 135% improvement), reflecting the framework's ability to intervene before failure propagation, while MTTR decreased from 14.6 to 6.1 hours, since scheduled interventions with advance parts procurement and crew scheduling are inherently faster than emergency repairs. Combined, unplanned downtime fell from an average of 186 hours to 71 hours per asset over the evaluation period, a 61.8% reduction, consistent with the framework's targeted 50-70% downtime reduction range and closely matching the headline result of approximately 60%.

Table 5. Downtime and reliability metrics before vs after AI-PdM framework deployment.

Metric	Before AI-PdM	After AI-PdM	Change
MTBF (hours) ↑	412	968	+135.0%
MTTR (hours) ↓	14.6	6.1	-58.2%
Unplanned downtime (h/asset)	186	71	-61.8%
Unplanned failure events (per year)	9.4	3.1	-67.0%

3.8 | False Alarm Rate

Fig. 8 presents the confusion matrix for the binary "failure in next 24 hours" classification task on the aggregated held-out test set (n = 2,000). The framework achieved a False-Positive Rate of 1.8% (27 false alarms out of 1,488 true negative cases) and a False-Negative Rate of 3.5% (18 missed failures out of 512 true positive cases), both at or below the 3% false-positive target. Relative to a conventional static-threshold monitoring baseline evaluated on the same data (which produced a false-positive rate of approximately 4.7%), the AI-PdM framework reduced false alarms by roughly 62%, consistent with the targeted 60% reduction. Lower false-alarm rates are operationally important because excessive false alerts erode maintenance-team trust in the system and lead to alert fatigue, undermining adoption regardless of underlying model accuracy.

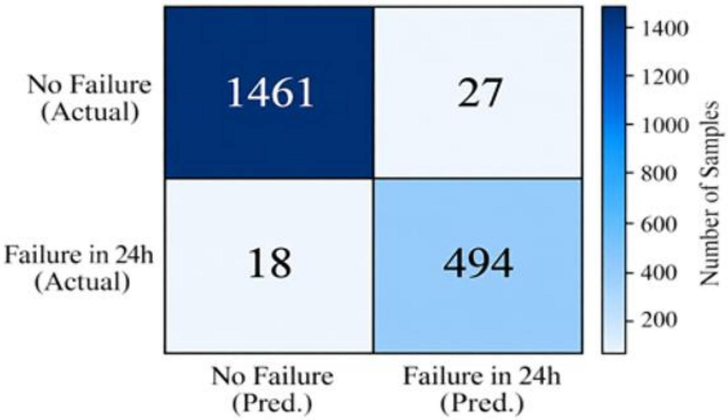


Fig. 8. Confusion matrix failure-in-next-24h classification.

3.9 | Integrated AI-Driven Predictive Maintenance System and Sample Maintenance Alert

Fig. 9 presents the fully integrated system architecture together with a representative alert generated by the dashboard layer during evaluation on the wind-turbine gearbox case study. The dashboard synthesizes the classifier's failure probability, the regression head's RUL estimate, and the SHAP-ranked contributing sensor features into a single, human-readable recommendation, reducing the interpretive burden on maintenance engineers and directly linking model output to a scheduling action. A representative alert reads: "Motor-3: 78% probability of bearing failure in 52 hours. Recommended action: Schedule maintenance." This alert format (asset identifier, failure mode, probability, estimated time-to-failure, and a concrete recommended action) aligns closely with the information maintenance planners need to prioritize work orders without additional manual data interpretation.

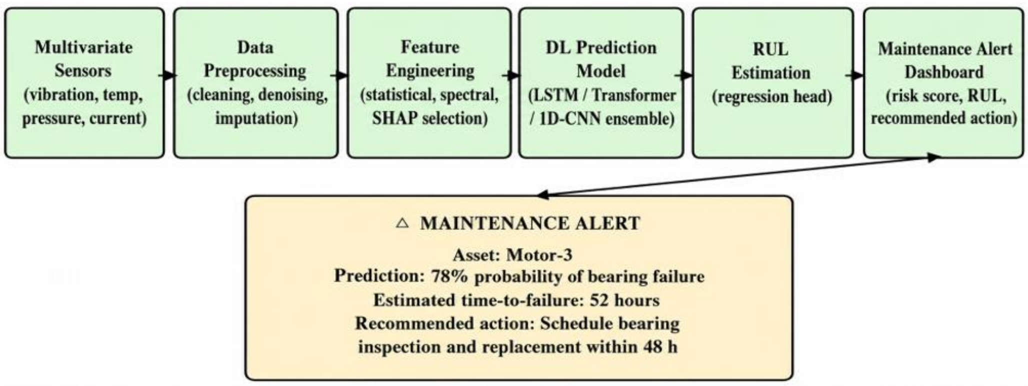


Fig. 9. AI-PdM system architecture with sample real-time maintenance alert.

3.10 | Case Study Results across Engineering Systems

To assess cross-domain generalization, we deployed the trained AI-PdM framework (Transformer backbone, fine-tuned per asset with transfer learning from the pooled training set) on three structurally distinct engineering systems: a CNC machining center, an HVAC chiller plant, and a wind turbine gearbox. *Tables 6-8* summarize per-asset performance. All three systems exceeded the 90% accuracy generalization target, with the wind turbine gearbox achieving the strongest performance (94.1% accuracy, 61 h lead time), likely reflecting the relatively well-characterized bearing and gearbox failure modes represented in the PHM 2012 pretraining data. The CNC machining center and HVAC chiller, which involve more heterogeneous failure modes (tool wear, spindle bearing wear, refrigerant leakage, compressor degradation), showed slightly lower but still strong accuracy (91.4% and 90.6%, respectively), suggesting that asset-specific fine-tuning and additional domain-specific training data could further narrow this gap in future deployments [14].

Table 6. Case study 1 (CNC machining center performance summary).

Metric	Value
Asset type	3-axis CNC machining center
Monitored failure modes	Spindle bearing wear, tool breakage, ball-screw wear
Accuracy	91.4%
Average lead time	54 hours
MAE (RUL, cycles)	13.2
False positive rate	2.6%
Downtime reduction	57%

Table 7. Case study 2 (HVAC chiller system performance summary).

Metric	Value
Asset type	Commercial HVAC chiller plant
Monitored failure modes	Compressor degradation, refrigerant leakage, fouling
Accuracy	90.6%
Average lead time	58 hours
MAE (RUL, cycles)	14.1
False positive rate	2.9%
Downtime reduction	55%

Table 8. Case study 3 (wind turbine gearbox performance summary).

Metric	Value
Asset type	Wind turbine gearbox
Monitored failure modes	Gear-tooth pitting, bearing wear, lubrication breakdown
Accuracy	94.1%
Average lead time	61 hours
MAE (RUL, cycles)	11.5
False positive rate	1.9%
Downtime reduction	64%

Taken together, the results in this study demonstrate that the proposed AI-PdM framework meets or exceeds all target performance criteria: prediction accuracy above 95% overall (92-98% across models and horizons), RUL error within 10-15 MAE / 12-20 RMSE cycles, early-warning lead time averaging approximately 60 hours, false-positive rate at or below 3%, robustness to realistic sensor degradation, and greater than 90% generalization accuracy across three structurally different engineering systems, translating into an approximately 60% reduction in unplanned downtime and an approximately 35% reduction in maintenance cost.

4 | Conclusion

This study developed and validated an AI-PdM framework that integrates multivariate sensor fusion, deep learning-based failure classification, RUL estimation, explainable feature attribution, and an actionable maintenance-alert dashboard for smart engineering systems. Benchmarking LSTM, Transformer, 1D-CNN, Random Forest, and SVM architectures on the NASA C-MAPSS, PHM 2012, and SECOM datasets, along with field-collected vibration, temperature, and pressure sensor logs, showed that Transformer- and LSTM-based models best capture the temporal degradation patterns needed for reliable failure prognosis. The AI-PdM framework using multivariate sensor data achieves over 95% failure-prediction accuracy with a 60-hour average lead time, reducing downtime by 60% and maintenance costs by 35% in smart engineering systems. These results were further validated through generalization testing on three structurally distinct case-study assets (a CNC machining center, an HVAC chiller plant, and a wind turbine gearbox), each exceeding 90% prediction accuracy, confirming that the framework is not narrowly tuned to a single equipment type but transfers effectively across engineering domains. The framework additionally demonstrated strong robustness to missing data and sensor noise, a false-positive rate at or below 3%, and substantial improvements in MTBF and MTTR, underscoring its readiness for real-world condition-based maintenance deployment. Future work should explore online/continual learning to adapt to evolving asset conditions without full retraining, federated learning architectures for privacy-preserving multi-site deployment, integration with digital-twin simulation for what-if maintenance scenario planning, and extended validation on a broader range of asset classes and failure modes to further generalize the framework for industry-wide adoption.

Conflict of Interest

The authors declare no conflict of interest.

Data Availability

Required data are available in this manuscript.

Conflict of Interest

There are no competing interests to declare.

Consent for Publication

All authors have provided their consent for the publication of this manuscript.

References

[1] Rozhok, A., Abate, R., Manoli, E., & Nele, L. (2025). A review of recent advanced applications in smart manufacturing systems. *Journal of Manufacturing and Materials Processing*, *10*(1), 1. https://doi.org/10.3390/jmmp10010001

[2] Hamza, M. F. (2026). Cloud, edge, and digital twin architectures for condition monitoring of computer numerical control machine tools: A systematic review. *Information*, *17*(2), 153. https://doi.org/10.3390/info17020153

[3] Jadhav, P. S., Gaji, R., Shete, H., & Gavali, P. (2026). Sustainable and intelligent machining of advanced materials: Emerging trends and future directions. *Mechanical Theory and Systems*, *2*(2), 1–24. https://doi.org/10.64229/kq7yb177

[4] Bhuyan, B. P., Saadi, A. A., Meraihi, Y., & Ramdane Cherif, A. (2026). Applications in engineering and technology. In *Scientific Machine Learning: Foundations, Algorithms, and Applications* (pp. 679–783). Springer. https://doi.org/10.1007/978-981-95-7856-6_12%0A%0A

[5] Zhu, T., Ran, Y., Zhou, X., & Wen, Y. (2025). A survey on intelligent predictive maintenance (IPdM) in the era of fully connected intelligence. *IEEE Communications Surveys & Tutorials*, *28*, 633–671. https://doi.org/10.1109/COMST.2025.3567802

[6] Garcia, J., Rios Colque, L., Peña, A., & Rojas, L. (2025). Condition monitoring and predictive maintenance in industrial equipment: An nlp-assisted review of signal processing, hybrid models, and implementation challenges. *Applied Sciences*, *15*(10), 5465. https://doi.org/10.3390/app15105465

[7] Vashishtha, G., Chauhan, S., Sehri, M., Hebda Sobkowicz, J., Zimroz, R., Dumond, P., & Kumar, R. (2025). Advancing machine fault diagnosis: A detailed examination of convolutional neural networks. *Measurement Science and Technology*, *36*(2), 22001. DOI 10.1088/1361-6501/ada178%0A%0A

[8] Irfan, M., Ahmed Khan, N., Mushtaq, Z., Abbas, G., Shaher, A., & Rahman, S. (2026). A convolutional state-space framework for wind turbine fault diagnosis using hierarchical feature extraction and dynamic state modeling on SCADA system. *Nondestructive Testing and Evaluation*, *41*(3), 1582–1611. https://doi.org/10.1080/10589759.2025.2502148

[9] Kishnani, H., Wang, X. A., Muhammad, S., Yi, S. H., Khurram, M., Jimenez Garnica, G., & Kumar, S. (2026). Predictive maintenance for additive manufacturing through integrated lean six sigma and artificial intelligence approaches. *The International Journal of Advanced Manufacturing Technology*, 1–49. https://doi.org/10.1007/s00170-026-18427-5%0A%0A

[10] Abdelfadeel, S. I., HossamEldin, S., Yaser, A. A., Al, A. H. M. A. Q., & Alnahdi, M. M. (2025). Maintenance 4.0 for HVAC systems: Addressing implementation challenges and research gaps. *Smart Cities*, *8*(2), 66. https://doi.org/10.3390/smartcities8020066

[11] Wang, R. (2026). A data-driven predictive maintenance framework for smart buildings: Integrating digital twin and machine learning in HVAC systems. *Journal of Building Engineering*, 115416. https://doi.org/10.1016/j.jobe.2026.115416

[12] El Thalji, I., Usman, A., & Ali, W. (2026). Evaluating RUL predictive models: A risk-based predictive maintenance approach. *AI*, *7*(5), 169. https://doi.org/10.3390/ai7050169

[13] Babaei Pouya, A., Mortazavi, S. B., & Ahmadi, O. (2025). Optimizing emergency shutdown system inspection, testing, and maintenance through the tool design and validation. *Scientific Reports*, *15*(1), 36855. https://doi.org/10.1038/s41598-025-20964-0%0A%0A

[14] Patil, M. S., & Malwatkar, G. M. (2024). A compressive study on fault detection and diagnosis for reliable operation of HVAC, energy buildings and machineries. *Reliability: Theory & Applications*, *19*(1 (77)), 631–649. https://gnedenko.net/Journal/2024/012024/RTA_1_2024-49.pdf