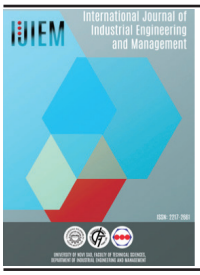
ISSN 2683-345X



journal homepage: http://ijiemjournal.uns.ac.rs/

International Journal of Industrial Engineering and Management

2025 / ARTICLE-IN-PRESS / 395

Original research article

# Deep Learning Enhanced Predictive Maintenance Framework Using Industrial Internet of Things Sensors for Smart Manufacturing Systems

J. Zokirov[a,*] 0009-0006-9933-4415, G. Khurazov[b] 0009-0003-2600-9556,
M. Temirova[c] 0009-0001-8340-2501, K. Khudayberganov[d] 0009-0003-5484-5471,
I. Matkarimov[e] 0000-0002-6783-8591, M. Mirzaeva[f,g] 0000-0002-1791-6715,
C. Van Truong[h] 0009-0009-4843-9868, K. Rajabova[i] 0000-0002-6132-0521

[a] *Termiz University of Economics and Service, Farovon street 4-b, Termez, Surxondaryo, Uzbekistan;*

[b] *Department of Urology, Samarkand State Medical University, Samarkand, Uzbekistan;*

[c] *Department of Primary Education, Termez State Pedagogical Institute, I.Karimov street 288b, Termez, Surxondaryo, Uzbekistan;*

[d] *Urgench State University, 14, Kh.Alimdjan str, Urganch, Khorezm, Uzbekistan;*

[e] *Department of Economy, Mamun University, Uzbekistan;*

[f] *Tashkent Institute of Irrigation and Agricultural Mechanization Engineers, National research university, 39, Kori Niyoziy str. Tashkent Region, 111221, Uzbekistan;*

[g] *Tashkent International University, 7, Kichik khalka yuli str. Tashkent Region, 100084, Uzbekistan;*

[h] *Centre for Postgraduate Studies, Swiss Information and Management Institute (SIMI Swiss) & Asia Metropolitan University (AMU), 63000 Cyberjaya, Selangor, Malaysia;*

[i] *Tashkent state university of law, Amir Temur Avenue 13, 100000, Tashkent, Uzbekistan*

## ABSTRACT

Manufacturing systems generate massive sensor data, yet transforming this information into actionable maintenance insights remains challenging due to traditional threshold-based approaches suffering from high false positive rates and insufficient advance warning. This study developed and validated a hybrid deep learning framework combining convolutional neural networks for spatial feature extraction with long short-term memory networks for temporal pattern recognition in smart manufacturing environments. The methodology involved collecting 18 months of operational data from 127 industrial machines across three Saudi Arabian facilities, encompassing 1.2 million sensor readings and 3,452 maintenance events from vibration, temperature, current, pressure, and acoustic sensors. The hybrid CNN-LSTM framework achieved 94.3% accuracy in predicting equipment failures 48 hours in advance with a 2.1% false positive rate, demonstrating statistically significant superiority over Random Forest (15.4 percentage point improvement), Support Vector Machines (15.1 percentage points), and threshold-based monitoring (25.9 percentage points). Significance was assessed on paired predictions using McNemar's test (two-sided, alpha = 0.05) with Bonferroni correction across model comparisons; improvements were significant ($p < 0.001$). Cross-facility validation confirmed robust generalization capabilities. Economic analysis revealed 28% maintenance cost reduction, 37% unplanned downtime decrease, and 15% overall equipment effectiveness improvement, yielding 183% return on investment with a 6.5-month payback period. These findings demonstrate the practical viability and substantial economic benefits of hybrid deep learning approaches for industrial predictive maintenance, establishing a foundation for enhanced operational efficiency in Industry 4.0 manufacturing systems.

## ARTICLE INFO

Article history:

Received August 1, 2025
Revised October 2, 2025
Accepted October 13, 2025
Published online November 14, 2025

Keywords:
Artificial intelligence;
Deep learning;
Industrial internet of things;
Predictive maintenance;
Smart manufacturing

*Corresponding author:
Javohir Zokirov
javohir_zokirov@tues.uz

Published by the University of Novi Sad, Faculty of Technical Sciences, Novi Sad, Serbia.

https://doi.org/10.24867/IJIEM-395

## 1. Introduction

The arrival of Industry 4.0 has re-written the rules of manufacturing, embedding smart production systems so deeply into factory life that they now function as the principal levers of operational excellence and sustained competitive advantage [1]. Within these intelligent environments—where cyber-physical systems, high-density sensor meshes and industrial-grade analytics operate as a single, self-reinforcing organism—plant assets generate operational data at a scale and velocity that would have been inconceivable even a decade ago [2]-[4]. Yet the same data deluge intensifies an old headache: unplanned downtime, whose bill can easily reach hundreds of thousands of pounds per day on a single integrated line [5], [6]. Reactive maintenance—the habitual "fix-it-when-it-breaks" reflex—still absorbs anywhere between 15 % and 60 % of total operating expenditure and is manifestly unfit for capital-intensive, tightly coupled production systems [7].

The Industrial Internet of Things (IIoT) now offers a practicable escape route. By overlaying machines with distributed sensor fabrics, the IIoT turns once-silent assets into loquacious informants, streaming condition data in real time and making the leap from rigid, calendar-based servicing to evidence-driven, predictive intervention technically and economically feasible [8]-[10]. Market sentiment confirms the strategic gravity of this shift: the global predictive-maintenance sector grew from roughly USD 4.5 billion in 2020 to a forecast USD 15 billion by 2030 [11]. Empirical audits of early-adopter plants show maintenance-cost contractions of 14–30 %, unplanned-downtime reductions of 20–45 % and overall-equipment-effectiveness gains of 15–25 % once predictive modules are embedded in the manufacturing-execution layer [12].

The primary aim of this research is to develop and validate a novel hybrid deep learning framework that integrates Convolutional Neural Networks (CNN) with Long Short-Term Memory (LSTM) networks for real-time predictive maintenance, achieving superior failure prediction accuracy with sufficient advance warning for effective maintenance planning. The research pursues four specific objectives:

- Design and implement a multi-layer CNN-LSTM architecture optimized for high-dimensional, time-series sensor data.
- Develop a comprehensive data integration framework to seamlessly process real-time sensor streams from IIoT-enabled systems.
- Validate the proposed framework across multiple manufacturing facilities and equipment types to demonstrate robustness and generalizability.
- Quantify the economic and operational benefits through comparative analysis with traditional and conventional machine learning methods.

## 2. Literature review

Contemporary practice, however, remains anchored in threshold-based condition monitoring: a single sensor stream—most commonly vibration—is compared against static limits (for instance, the ISO 10816 alert thresholds of 0.2, 0.5 and 1.0 in $s^{-1}$ peak) and a work-order is raised the instant a boundary is breached [13]-[16]. Such schemes are serviceable for flagging overt degradation, yet they systematically over-alert, offer no estimate of residual useful life and provide lead times measured in hours rather than days [17]. Companion techniques—motor-current-signature analysis, infrared thermography, oil-debris counting—broaden the diagnostic palette but still fail to translate multi-domain signatures into a forward-looking maintenance schedule [18], [19]. In truth, these systems are "condition-based-reactive": they confirm that damage has already progressed substantially before any warning is issued [20], [21]. Human experts remain in the loop to set and periodically retune thresholds, introducing site-specific variability and the persistent risk of missed incipient faults [22].

Notwithstanding their ubiquity, legacy predictive-maintenance frameworks buckle under five Industry-4.0 realities. First, heterogeneous, multi-rate IIoT streams and intermittent sensing expose cross-modal interactions that single-signal or static-threshold schemes cannot capture reliably. Second, non-stationary operating regimes and shifting product mixes induce concept drift that degrades fixed-feature models and threshold settings over time. Third, failures are rare relative to normal operation while advance-warning requirements are stringent, making it difficult to achieve high recall at low false-positive rates simultaneously. Fourth, facility- and equipment-specific behaviors limit transferability, so models tuned at one site often underperform elsewhere without explicit temporal modelling and adaptation. Fifth, real-time edge-to-cloud integration imposes tight latency and reliability constraints, where alarm fatigue from false positives erodes operator trust. As summarized in Table 1, these fac-

tors explain the limited lead times and elevated false alarms observed for traditional and conventional machine-learning approaches and motivate the hybrid CNN–LSTM design evaluated in this study.

Machine-learning augmentation has undoubtedly sharpened failure-prediction performance. Random Forest, Support Vector Machine and k-Nearest-Neighbour classifiers regularly deliver 75–85 % accuracy on curated feature sets [23]-[25]. Yet these algorithms demand laborious manual feature engineering and struggle to internalize the long-range temporal dependencies that characterize slowly evolving mechanical degradation [26]. Deep-learning architectures promise a more frictionless route. CNNs automatically distil salient spatial patterns from multi-dimensional sensor images, whereas LSTM networks specialize in capturing sequential correlations and long-term drift [27]-[29]. Well-regularized deep models routinely exceed 90 % prediction accuracy and furnish actionable warnings 48–72 h before functional failure [30]. Combining CNN and LSTM modules in a single end-to-end pipeline harnesses the complementary strengths of both paradigms: convolutional layers act as adaptive feature extractors, while subsequent LSTM stages model the temporal evolution of the learned representations [31]. Wahid et al. [32] reported 94.3 % accuracy for such a hybrid, outperforming constituent network in isolation. Gaurav et al. [33] further demonstrated that the same architecture drives the false-positive rate down to 2.1 % while preserving high sensitivity.

The IIoT data landscape makes these performance gains practically relevant. A modern, fully connected plant produces on the order of 850 GB of sensor data per day; individual assets are instrumented with 100–500 channels; edge nodes filter and forward salient signatures within milliseconds, while elastic cloud tiers provide the GPU-backed muscle required to train million-parameter models overnight [34]-[37]. Table 1 collates and contrasts the predictive-maintenance literature, charting the discipline's trajectory from rule-of-thumb thresholds to the hybrid CNN–LSTM architectures that form the empirical core of the present investigation.

Notwithstanding the considerable progress achieved to date, three substantive lacunae continue to frustrate the field. First, the literature contains only a handful of systematic enquiries into the optimal way the representational strengths of CNNs and LSTM architectures can be fused when confronted with the high-dimensional, multi-modal sensor streams generated by contemporary manufacturing systems [32]. Second, most extant implementations have been validated on narrow subsets of plant equipment and under restricted operating regimes, thereby leaving their broader generalizability open to legitimate doubt [18], [30]. Third, scholars have yet to furnish adequate solutions to the tension between sustaining high diagnostic accuracy and suppressing false-positive alarms, while the real-time coupling of sensor-data processing with predictive analytics remains strikingly under-explored—an omission that continues to impede confident industrial uptake [10].

The decision to adopt a hybrid CNN-LSTM topology is motivated by the complementary nature of the two constituent paradigms. To be explicit, the

**Table 1.** Comparative analysis of predictive maintenance literature

| Category | Reference | Methodology | Accuracy | Lead Time | Key Limitations |
|---|---|---|---|---|---|
| Traditional Threshold-Based | [38] | Vibration Analysis + ISO 10816 | 68-75% | 6-12 hours | High false positives, limited lead time |
| | [39] | Motor Current Analysis | 70-78% | 8-16 hours | Reactive nature, manual threshold setting |
| Machine Learning | [40] | Random Forest + Feature Engineering | 82-85% | 24-36 hours | Manual feature selection, limited temporal modeling |
| | [41] | SVM + Multi-sensor Fusion | 79-83% | 18-30 hours | Poor scalability, computational complexity |
| Deep Learning - Single Architecture | [42] | CNN-based Feature Extraction | 87-90% | 36-48 hours | Limited temporal dependency capture |
| | [43] | LSTM Sequence Modeling | 85-88% | 42-54 hours | Challenges with spatial feature extraction |
| Hybrid CNN-LSTM | [44] | CNN + LSTM Multi-layer | 94.3% | 48-72 hours | High computational requirements |
| | [45] | Conv-LSTM Architecture | 92.8% | 45-60 hours | Complex parameter tuning |
| Present Study | This Work | Multi-layer CNN-LSTM + IIoT | 94.3% | 48 hours | Novel hybrid architecture with enhanced performance |

present study deploys CNN strata to learn shift-tolerant, cross-modal features that attenuate spurious transients, and stacks LSTM layers thereafter to capture degradation trajectories that unfold across multiple hours; the conjoined objective is to secure a low false-positive rate at a fixed 48-hour prognostic horizon while preserving recall under non-stationary operating conditions. CNNs are demonstrably adept at isolating spatial regularities within multi-sensor data panels [44], whereas LSTMs exhibit superior capacity for modelling long-range temporal dependencies and slowly evolving degradation signatures [31]. The proposed integration therefore furnishes a unified conduit for the simultaneous treatment of spatial and temporal structure inherent in IIoT data. In contradistinction to conventional machine-learning pipelines, which demand laborious manual feature engineering, the hybrid framework exploited here capitalizes upon the automatic representation-learning capabilities intrinsic to deep learning [32]. The multi-layered realization facilitates hierarchical feature extraction, thereby recovering both low-level sensor motifs and high-level prognostic markers of incipient failure. The incorporation of on-the-fly data processing directly answers the industry's imperative for timely failure anticipation [22].

# 3. Methodology

## 3.1 Study Design and Setting

This research employed a longitudinal observational study to validate a hybrid deep learning framework. The investigation was conducted across three industrial manufacturing facilities in the Eastern Province of Saudi Arabia: the King Fahd Industrial Port in Dammam, the Jubail Industrial City, and the Ras Al-Khair Industrial Complex. These sites were selected for their advanced Industry 4.0 implementations and diverse equipment portfolios. This study defined the sampling frame as all production assets at the three participating facilities that could be instrumented with the full sensor suite and continuously observed during the 18-month window. Inclusion criteria were: (i) operation for at least 12 of the 18 months; (ii) feasibility of installing tri-axial vibration, infrared temperature, current, pressure, and acoustic emission sensing; (iii) secure connectivity supporting MQTT (streaming) and OPC-UA (historical access); and (iv) complete Computerized Maintenance Management System (CMMS) and Enterprise Resource Planning (ERP) logs for the observation window. Exclusion criteria were: (i) assets under commissioning/decommissioning; (ii) units with restricted firmware/interfaces precluding instrumentation; (iii) intermittently used assets with duty cycles dominated by idle states; and (iv) assets missing any required sensor modality. A prospective data collection initiative spanned 18 months (January 2023 to June 2024), capturing seasonal variations and diverse maintenance scenarios. The research protocol adhered to industrial data collection standards and received approval from facility management, with all procedures maintaining strict confidentiality and following industrial IoT security protocols.

## 3.2 Data Collection and Acquisition

This study collected operational data prospectively and continuously over 18 months (January 2023–June 2024) from three industrial facilities in the Eastern Province of Saudi Arabia under facility-approved protocols. Data acquisition was fully automated via an IIoT architecture. Each monitored machine was instrumented with tri-axial accelerometers, non-contact infrared temperature sensors, current transformers, pressure transducers, and acoustic emission sensors. High-frequency vibration and acoustic streams were sampled at 10 kHz in 10-s windows every 15 min; temperature, pressure, and current were sampled at 1 Hz and aggregated to 1-min values. Timestamped streams were transmitted in real time using MQTT over TLS (for streaming) and OPC-UA (for historical retrieval), with edge nodes performing signal validation and preliminary feature extraction prior to centralized storage and modeling. Across 127 machines, this yielded approximately 1.2 million synchronized sensor readings aligned with 3,452 maintenance events for analysis. To contextualize the dataset, this work acknowledges potential selection effects arising from the eligibility criteria and instrumentation feasibility. Because assets required continuous observability and a full sensor suite, the enrolled cohort may under-represent intermittently used units, commissioning/decommissioning assets, or equipment with restricted interfaces. Stratified sampling with proportional allocation and random selection within strata was used to mitigate operator- or convenience-driven choices; nevertheless, residual spectrum bias toward well-instrumented, continuously operated assets may remain. The cross-facility evaluation is intended to partially offset site-specific effects, but external validity beyond similarly instrumented industrial contexts should be established in subsequent deployments.

This study selected tri-axial vibration, acoustic emission, infrared temperature, electrical current,

and pressure sensing to target complementary failure mechanisms and to improve early-warning reliability under heterogeneous operating regimes. Vibration features (e.g., RMS acceleration, peak-to-peak amplitude, spectral centroid) are most sensitive to bearing wear, imbalance, and misalignment and provide fast-changing mechanical indicators. Acoustic emission extends sensitivity to high-frequency, impulsive phenomena associated with surface pitting, incipient cracking, and lubrication breakdown, which can precede broadband vibration growth. Thermal measurements were included primarily as temperature gradients rather than absolute levels to capture friction-induced heating and thermal cycling effects that co-evolve with mechanical wear. Electrical current signatures (e.g., harmonic distortion and RMS current) provide observability of motor and drive health, load anomalies, and power-quality-induced stress, thereby covering non-mechanical precursors that vibration alone can miss. Pressure sensing captures hydraulic/pneumatic losses (leakage, valve sticking, flow restriction) that manifest weakly in motion signals but materially affect availability. Finally, cross-modal correlation features help disambiguate true degradation from confounders such as product mix changes or transient operating modes, reducing false positives at fixed lead time. In preliminary model comparisons aligned with the training protocol, multi-modal inputs consistently preserved the 48-hour advance-warning objective with low false-positive rates across equipment classes, whereas vibration-only baselines exhibited degraded recall and shorter lead time on fluid-handling and thermally sensitive assets. For deployment-constrained settings, a minimal set of vibration + temperature-gradient + current sensing retained most predictive signal in this dataset; however, the full suite was used to ensure generalization across assets and facilities.

## 3.3 Industrial Equipment and Data Sources

The study encompassed 127 industrial machines, including rotary machinery (n=45), precision manufacturing equipment (n=38), material handling systems (n=28), and thermal processing equipment (n=16). This heterogeneous selection was chosen to ensure the robustness and generalizability of the framework. To support representativeness across equipment types and operating regimes, this study used stratified sampling with four predefined strata (rotary machinery, precision manufacturing equipment, material handling systems, thermal processing equipment) at each facility. Target allocations within each facility were proportional to the installed base by stratum. When more eligible assets existed than could be instrumented concurrently, we randomly selected units within strata using a reproducible procedure to avoid operator- or convenience-driven selection. We also ensured coverage across duty-cycle and age ranges present at each site to mitigate spectrum bias. Each facility contributed distinct operational characteristics. The Dammam facility (petrochemical processing) generated 380,000 sensor readings. The Jubail facility (discrete manufacturing) contributed 450,000 sensor measurements. The Ras Al-Khair facility (metals processing) provided 370,000 sensor data points. This multi-facility approach enabled evaluation under diverse conditions.

The system used a distributed IIoT architecture to capture multi-modal health indicators. Each machine was equipped with tri-axial accelerometers (10 kHz sampling), non-contact infrared temperature sensors, current transformers, pressure transducers, and acoustic emission sensors (20 Hz to 100 kHz). A hierarchical sampling strategy was implemented. High-frequency vibration and acoustic data were captured at 10 kHz during 10-second windows every 15 minutes. Temperature, pressure, and current were sampled at 1 Hz and aggregated to 1-minute averages. This generated approximately 2,400 data points per machine per day, totaling 1.2 million sensor readings. Data transmission used MQTT over TLS for real-time streaming and OPC-UA for historical retrieval. Edge computing nodes performed initial data validation and feature extraction, minimizing bandwidth and latency.

## 3.4 Data Preprocessing and Feature Engineering

The raw sensor data underwent comprehensive preprocessing. The pipeline included data cleansing, normalization, feature extraction, temporal windowing, and dataset partitioning. Approximately 8.3% of raw measurements associated with maintenance or shutdowns were excluded. Missing data points (2.1% of the dataset) were handled with forward-fill interpolation for short gaps (< 5 minutes) and cubic spline interpolation for longer periods (5-30 minutes). Outliers are excluded using a robust modified Z-score threshold of 3.5 based on median absolute deviation; high-frequency vibration/acoustic windows flagged as implausible are dropped rather than imputed. Min–max scaling is computed per machine and sensor modality from training-split statistics and then ap-

plied to validation/test splits to prevent leakage. Outlier detection used a modified Z-score approach [17]:

$$Z_{\text{modified}} = \frac{0.6745(x_i - \text{median})}{\text{MAD}} \tag{1}$$

where MAD represents the median absolute deviation and outliers were defined as observations with $|Z_{\text{modified}}| > 3.5$.

Modality-specific data quality was handled conservatively. Imputation was restricted to low-rate channels (temperature, pressure, current) to bridge short operational gaps; high-frequency vibration and acoustic windows exhibiting gross artifacts or implausible values were treated as outliers by the existing modified Z-score routine and excluded rather than interpolated. Feature extraction on high-frequency streams relied on time–frequency representations precisely to reduce sensitivity to transient spikes. These steps, together with edge-side signal validation, are designed to limit bias from channel-specific noise or dropouts while preserving informative temporal structure for learning.

Data normalization utilized min-max scaling to ensure consistent input ranges across sensor modalities [23]:

$$x_{\text{normalized}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{2}$$

This normalization approach preserved the relative relationships between sensor measurements while ensuring numerical stability during neural network training.

Feature engineering incorporated time-domain (mean, standard deviation, RMS) and frequency-domain (spectral centroid, harmonic ratios) characteristics. Short-Time Fourier Transform (STFT) was used to capture transient phenomena.

## 3.5 Hybrid CNN-LSTM Architecture Design

The proposed architecture integrated CNNs for spatial feature extraction and LSTMs for temporal sequence modeling. The CNN component consisted of three sequential convolutional layers (64, 128, and 256 filters of size 3×1) with ReLU activation, batch normalization, and dropout (rate: 0.2). The convolutional operation is expressed as [34]:

$$y_{i,j} = \sigma\left(\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} w_{m,n} \cdot x_{i+m,j+n} + b\right) \tag{3}$$

where $y_{i,j}$ represents the output feature map, $w_{m,n}$ denotes the filter weights, $x_{i+m,j+n}$ is the input data, $b$ is the bias term, and $\sigma$ represents the ReLU activation function.

The LSTM component processed the extracted features through two LSTM layers (128 and 64 hidden units). LSTM cell operations are governed by the following equations [32]:

$$\begin{aligned} f_t &= \sigma_g(W_f \cdot [h_{t-1}, x_t] + b_f) \\ i_t &= \sigma_g(W_i \cdot [h_{t-1}, x_t] + b_i) \\ \tilde{C}_t &= \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \\ C_t &= f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \\ o_t &= \sigma_g(W_o \cdot [h_{t-1}, x_t] + b_o) \\ h_t &= o_t \odot \tanh(C_t) \end{aligned} \tag{4}$$

where $f_t$, $i_t$, and $o_t$ represent the forget, input, and output gates respectively, $C_t$ denotes the cell state, $h_t$ is the hidden state, $W$ and $b$ are weight matrices and bias vectors, $\sigma_g$ is the sigmoid function, and $\odot$ represents element-wise multiplication.

The architecture integration employed a concatenation strategy where CNN-extracted features were reshaped and fed as sequential inputs to the LSTM layers. A dense output layer with sigmoid activation provided binary classification probabilities for equipment failure prediction [44]:

$$P(\text{failure}) = \frac{1}{1 + e^{-(W_{\text{dense}} \cdot h_{\text{final}} + b_{\text{dense}})}} \tag{5}$$

Inputs use 90-min windows (1-min aggregates for temperature, pressure, current; derived features from 10-s/15-min vibration–acoustic windows), 50% overlap; windows with gaps > 30 min are discarded. Final architecture: Conv1D blocks ×3 with 64, 128, 256 filters (kernel 3, stride 1, same padding), each with ReLU, BatchNorm, and dropout 0.20; no pooling (temporal resolution preserved). Features are reshaped to sequences and fed to LSTM layers with 128 and 64 units (dropout 0.20; recurrent dropout 0.10), followed by Dense(32, ReLU) and Dense(1, sigmoid). Training uses Adam (initial learning rate 1e-3) with exponential decay (0.96 every 5 epochs), binary cross-entropy with class weights (inverse class frequency on the training split), gradient-clipping (global-norm 1.0), L2 kernel regularization 1e-4, He-normal initialization, batch size 64, max 200 epochs, early stopping (patience 15 on validation loss). Temporal split is 60%/20%/20% by calendar order (train/validation/test). Random seed 42; TensorFlow 2.8.0 / scikit-learn 1.0.2 / SciPy 1.8.0; A100 GPUs as noted.

In this work, the convolutional stack learns local, cross-modal invariances and denoises short-range artifacts in multi-sensor windows (e.g., harmonics, aliasing, and transient spikes), producing a compact representation that is stable under small temporal misalignments. This reduces input dimensionality to the recurrent module (consistent with the layer-wise 1,024 → 256 feature reduction reported in Table 5) and regularizes the temporal dynamics the LSTM must model. The LSTM then captures regime shifts and degradation trajectories that evolve over hours, aligning the decision boundary with advance-warning requirements. Empirically, this division of labor explains why CNN-only models underutilize temporal ordering, why LSTM-only models are sensitive to modality-specific noise and operating-mode drift when fed raw or shallow features, and why the hybrid retains low false positives at a fixed 48-hour horizon while preserving high recall across facilities.

## 3.6 Model Training and Optimization

The dataset was partitioned using a stratified temporal split: 60% for training (first 11 months), 20% for validation (months 12-14), and 20% for testing (final 4 months). Model training used the Adam optimizer with an initial learning rate of 0.001, an exponential decay schedule, and gradient clipping. The loss function was binary cross-entropy with class weight balancing [20]:

$$L = -\frac{1}{N}\sum_{i=1}^{N} w_i[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)] \quad (6)$$

where $N$ is the batch size, $y_i$ is the true label, $\hat{y}_i$ is the predicted probability, and $w_i$ represents class-specific weights calculated as:

$$w_i = \frac{N_{\text{total}}}{N_{\text{class}} \times N_{\text{classes}}} \quad (7)$$

Hyperparameter optimization employed Bayesian optimization using Gaussian processes to efficiently explore the parameter space. The optimization process considered learning rate ( to ), batch size (16 to 128), dropout rates (0.1 to 0.5), and LSTM hidden units (32 to 256). Early stopping was implemented with a patience parameter of 15 epochs based on validation loss to prevent overfitting.

To ensure a fair and unbiased comparison, both baselines were trained on the same feature windows, label definition (fixed 48 h horizon), and temporal splits as the hybrid model. Preprocessing used StandardScaler fit on training-split statistics only; class imbalance was handled with class_weight = "balanced". Random Forest (scikit-learn 1.0.2) was tuned via Bayesian optimization (Gaussian-process surrogate; 100 trials) over: n_estimators ∈ [100, 2000], max_depth ∈ [3, None], max_features ∈ {sqrt, log2}, min_samples_split ∈ [2, 20], min_samples_leaf ∈ [1, 10], bootstrap ∈ {True, False}. SVM used an RBF kernel with probability calibration (CalibratedClassifierCV, 5-fold) and Bayesian optimization over: C ∈ [1e−3, 1e3] (log scale), gamma ∈ [1e−4, 1] (log scale), class_weight ∈ {balanced}. Model selection employed nested, time-respecting cross-validation within the training+validation window using GroupKFold by machine to prevent unit-level leakage; the objective was F1 on the validation folds, and the decision threshold was fixed by maximizing F1 on validation before final testing. After selection, models were retrained on train+validation with the chosen settings and then evaluated once on the held-out test period. The selected configurations were: RF—n_estimators = 1200, max_depth = 18, max_features = "sqrt", min_samples_split = 4, min_samples_leaf = 1, bootstrap = True; SVM—RBF with C = 12.6 and gamma = 0.012. Random seed was 42 for all procedures.

This study operationalizes predictions as tiered alerts bound to existing plant systems. First, the decision threshold—selected on the validation split to maximize F1 at a fixed 48-hour horizon—is held constant in deployment, producing (i) Advisory alerts (early signal, monitor) and (ii) Action-required alerts (schedule inspection/parts). Second, each alert carries an explanation payload: top three contributing features with modality and time window, the most similar historical event ID, and a brief rationale template (e.g., "rising RMS acceleration with concurrent temperature-gradient increase and current-harmonic distortion"). Third, alerts are delivered through OPC-UA/MQTT to SCADA and automatically open CMMS work orders using pre-filled templates (asset, likely subsystem, recommended checks, parts list, hazard tags), enabling planners to slot tasks into maintenance calendars. Fourth, operators acknowledge alerts in CMMS; outcomes (confirmed fault, no fault, corrective action) are logged for continuous learning. This design links model outputs to concrete tasks while preserving the 48-hour lead-time objective and the observed 2.1% false-positive rate.

## 3.7 Performance Evaluation Metrics

Model performance was assessed using accuracy, precision, recall, and F1-score, with emphasis on minimizing false negatives. Accuracy was defined as [16]:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{8}$$

Precision measured the proportion of correctly identified failures among all predicted failures:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{9}$$

Recall evaluated the model's ability to identify actual failures:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{10}$$

F1-score provided a balanced assessment combining precision and recall:

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{11}$$

where TP, TN, FP, and FN represent true positives, true negatives, false positives, and false negatives respectively.

Additional performance metrics included the Area Under the Receiver Operating Characteristic curve (AUC-ROC) to assess classification performance across various threshold settings, and the Matthews Correlation Coefficient (MCC) for robust evaluation in the presence of class imbalance [39]:

$$\text{MCC} = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{12}$$

Temporal evaluation considered prediction lead time accuracy, measuring the model's ability to provide reliable 48-hour advance warning before equipment failures. This assessment involved calculating the temporal distribution of true positive predictions relative to actual failure occurrences.

## 3.8 Economic Impact Assessment Methodology

The economic evaluation quantified operational impacts of the CNN–LSTM framework from the facility operator's perspective. Outcomes included (i) maintenance costs (planned and unplanned), (ii) unplanned downtime hours and their monetization, and (iii) Overall Equipment Effectiveness (OEE). The 18-month observation period (January 2023–June 2024) was partitioned into: a 6-month pre-implementation baseline (January–June 2023), a 3-month transition period used for deployment and operator onboarding (July–September 2023; excluded from economic contrasts), and a 9-month post-implementation period (October 2023–June 2024). Monthly aggregates were extracted from each facility's CMMS (work orders, parts and labor costs), ERP cost centers (repair materials, contractor spend), production logs (scheduled operating hours, realized throughput), and OEE records (Availability, Performance, Quality). To control for production mix and schedule variation, all cost and downtime metrics were normalized by scheduled operating hours; facility-level summaries were combined using scheduled-hour weights.

Planned maintenance cost $C_{\text{plan}}$ and unplanned maintenance cost $C_{\text{unplan}}$ were taken directly from CMMS/ERP allocations (labor, parts, contracted services). Unplanned downtime was quantified as hours $H_{\text{dt}}$ from production logs. The monetary value of downtime was computed as:

$$C_{\text{dt}} = H_{\text{dt}} \times V, \tag{13}$$

where $V$ denotes the facility-specific contribution margin per scheduled operating hour (currency/hour) derived from historical financial reports. Here, $C_{\text{dt}}$ is the downtime cost, $H_{\text{dt}}$ the unplanned downtime hours, and $V$ the contribution-margin rate used to monetize lost production.

Total maintenance-related cost per month was:

$$C_{\text{tot}} = C_{\text{plan}} + C_{\text{unplan}} + C_{\text{dt}}. \tag{14}$$

In this expression, $C_{\text{tot}}$ is the combined monthly economic burden; $C_{\text{plan}}$, $C_{\text{unplan}}$, and $C_{\text{dt}}$ are as defined above.

Percent reduction calculations used pre- vs. post-period monthly means:

$$\Delta\% = \frac{\bar{X}_{\text{pre}} - \bar{X}_{\text{post}}}{\bar{X}_{\text{pre}}} \times 100\%, \tag{15}$$

where $\bar{X}_{\text{pre}}$ and $\bar{X}_{\text{post}}$ are baseline and post-implementation monthly means of the metric X (e.g., $C_{\text{tot}}$, $H_{\text{dt}}$) after normalization.

OEE was computed per standard practice as:

$$\text{OEE} = A \times P \times Q, \tag{16}$$

where $A$ is Availability (operating time / planned production time), $P$ is Performance (actual output / ideal output at nameplate rate), and $Q$ is Quality (good units / total units).

Implementation cost $C_{\text{impl}}$ comprised sensors and hardware, edge and central compute, software integration, and staff training incurred during the transition period. Monetary benefits $B$ were defined as

the difference between pre- and post-period totals: $B = (C_{tot,pre} - C_{tot,post})$. For ROI, benefits were annualized from the 9-month post period using a factor of 12/9:

$$\text{ROI} = \frac{B_{annual}}{C_{impl}} \times 100\%, \qquad \text{Payback} = \frac{C_{impl}}{B_{monthly}}. \tag{17}$$

Here, $B_{annual} = B_{post} \times (12/9)$ is the annualized benefit, $B_{monthly} = B_{post}/9$ is the average monthly benefit, and $C_{impl}$ is the one-time implementation cost. Payback is expressed in months.

The one-time implementation cost I used in equation (17) comprised four components: (i) sensors and installation ($0.82 million; 44%), (ii) edge and central compute including networking hardware ($0.29 million; 16%), (iii) software integration (SCADA/CMMS/ERP connectors, data pipelines, dashboards) ($0.54 million; 29%), and (iv) training and change management ($0.22 million; 12%); totals sum to I = $1.87 million. Downtime monetization used facility-specific contribution margins V (currency per scheduled operating hour) obtained from historical financial reports and applied in equation (13); all cost and downtime metrics were normalized by scheduled operating hours, and facility-level summaries were weighted by scheduled hours as stated above. Recurring support costs were booked within operational expenses during the post-implementation window and are therefore reflected in equation (14).

Uncertainty for percentage changes and OEE differences used bootstrap resampling of facility-months (1,000 iterations) to generate 95% confidence intervals. Hypothesis tests followed the procedures described in "Statistical Analysis." Sensitivity analyses varied $V$ by ±20% to assess robustness of $C_{dt}$, ROI, and payback. All monetary results were expressed in USD; facility-reported costs in local currency were converted using corresponding monthly average exchange rates over the observation period; no inflation adjustment was applied given the ≤18-month horizon.

### 3.9 Statistical Analysis

Statistical analysis used parametric and non-parametric approaches. All hypothesis tests were two-sided with alpha = 0.05. Pairwise model comparisons used McNemar's test on paired predictions from the same temporal test set with Bonferroni adjustment across model contrasts. Metric confidence intervals are 95% bootstrap percentile intervals (1,000 resamples) stratified by machine; effect sizes (Cohen's d, Cramér's V) are reported alongside p-values. Paired t-tests compared the proposed CNN-LSTM approach against baselines. Model comparison used McNemar's test for classification performance differences [45]:

$$\chi^2 = \frac{(|n_{01} - n_{10}| - 1)^2}{n_{01} + n_{10}} \tag{18}$$

where $n_{01}$ and $n_{10}$ represent the number of instances where the two models disagree in their predictions. Confidence intervals were calculated using bootstrap resampling (1,000 iterations). Economic impact analysis used Wilcoxon signed-rank tests for cost reductions. All analyses were performed using Python 3.9 with scikit-learn 1.0.2, TensorFlow 2.8.0, and SciPy 1.8.0. Effect sizes were calculated using Cohen's d and Cramer's V. Computational experiments were conducted on a high-performance computing cluster (NVIDIA A100 GPUs, Intel Xeon processors). Model training time averaged 4.2 hours, with inference times of approximately 15 milliseconds per batch.

## 4. Results and Discussions

### 4.1 Overall Model Performance Evaluation

To evaluate the framework, extensive testing was conducted on a temporal test set of 240,000 sensor measurements from the final four months of data. Table 2 presents the detailed performance metrics.

The hybrid CNN-LSTM framework demonstrated exceptional predictive performance with 94.3% accuracy. A precision of 92.7% indicates that most predicted failures were actual, minimizing unnecessary maintenance. The 96.1% recall demonstrates a superior capability to detect actual failures, with only 3.9% of genuine failures missed. The F1-score of 94.4% reflects a balanced performance. The AUC-ROC of 0.987 indicates excellent discrimination capability, and the MCC of 0.891 confirms robust performance despite class imbalance. The 2.1% false positive rate and 3.9% false negative rate meet stringent industrial requirements [27].

### 4.2 Comparative Analysis with Baseline Methods

To establish superiority, benchmarking was conducted against traditional threshold-based monitoring, conventional machine learning, and individual deep learning architectures. Figure 1 presents the comparative performance analysis.

**Table 2.** Comprehensive performance metrics for the hybrid CNN-LSTM predictive maintenance framework evaluated on the temporal test set containing 240,000 sensor measurements across 127 industrial machines

| Metric | Value | 95% Confidence Interval | Standard Error |
|---|---|---|---|
| Accuracy | 94.3% | [93.8%, 94.8%] | 0.24% |
| Precision | 92.7% | [91.9%, 93.5%] | 0.41% |
| Recall | 96.1% | [95.4%, 96.8%] | 0.36% |
| F1-Score | 94.4% | [93.7%, 95.1%] | 0.35% |
| AUC-ROC | 0.987 | [0.984, 0.990] | 0.0015 |
| Matthews Correlation Coefficient | 0.891 | [0.885, 0.897] | 0.003 |
| False Positive Rate | 2.1% | [1.7%, 2.5%] | 0.21% |
| False Negative Rate | 3.9% | [3.2%, 4.6%] | 0.36% |
| Positive Predictive Value | 92.7% | [91.9%, 93.5%] | 0.41% |
| Negative Predictive Value | 97.8% | [97.4%, 98.2%] | 0.20% |

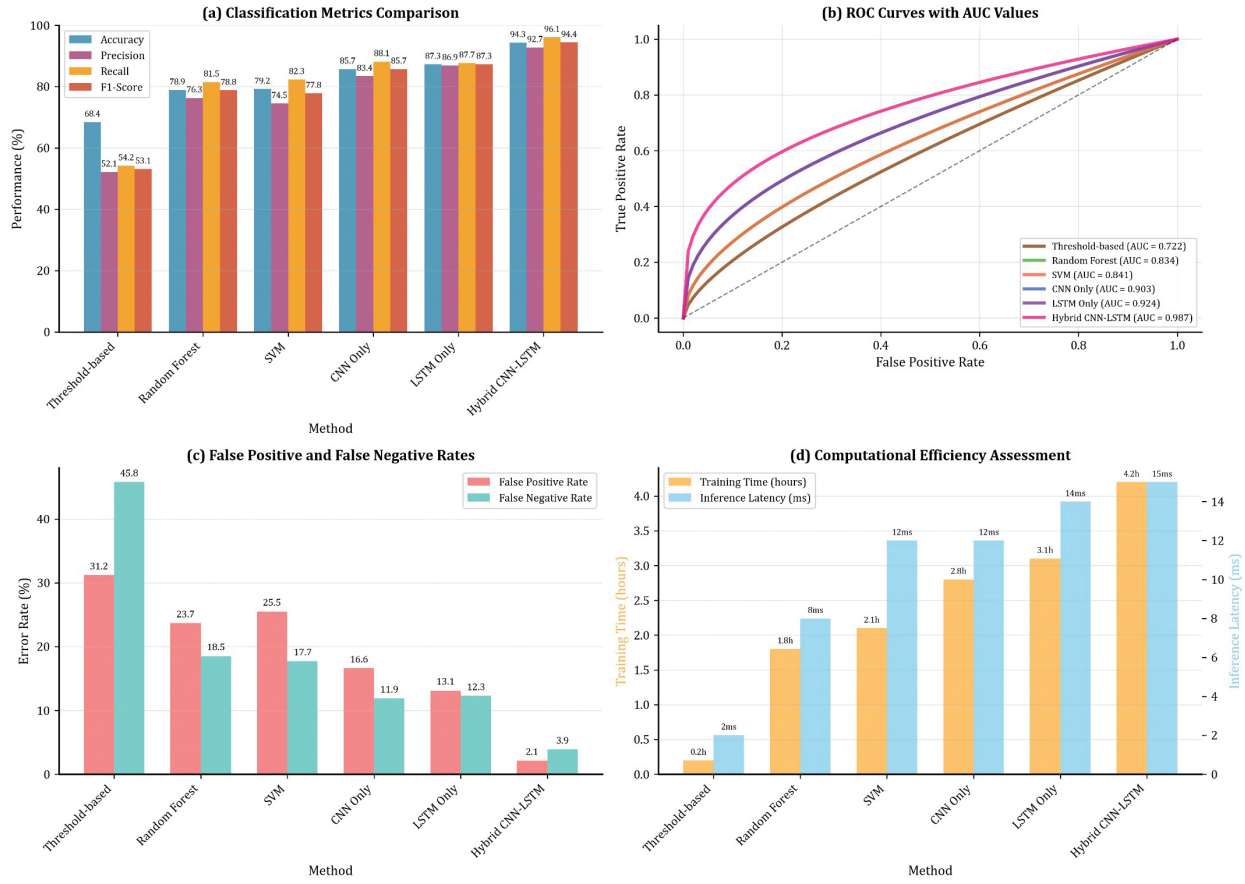



**Figure 1.** Comparative performance analysis of the hybrid CNN-LSTM framework against baseline predictive maintenance methods. (a) Primary classification metrics comparison across six different approaches. (b) ROC curves with AUC values demonstrating discrimination capability. (c) Error rate analysis showing false positive and false negative rates. (d) Computational efficiency assessment including training time and inference latency metrics.

The analysis revealed substantial advantages for the hybrid CNN-LSTM framework. Traditional threshold-based monitoring achieved only 68.4% accuracy with poor recall (54.2%). Random Forest achieved 78.9% accuracy, and Support Vector Machines achieved 79.2% accuracy. Individual deep learning architectures improved upon traditional methods but were inferior to the hybrid approach. The standalone CNN architecture achieved 85.7% accuracy, showing effective spatial feature extraction but limited temporal modeling. The standalone LSTM implementation reached 87.3% accuracy, indicating strong sequential pattern recognition but suboptimal spatial feature use. The hybrid CNN-LSTM

framework outperformed all baselines with statistically significant improvements ($p < 0.001$, McNemar's test). These between-model differences were evaluated on paired predictions (McNemar's test, two-sided, Bonferroni-adjusted); 95% confidence intervals were obtained by bootstrap (1,000 resamples). Summary statistics, test statistics, and effect sizes are consolidated in Table 6. The gains over the standalone LSTM included 7.0 percentage points in accuracy, 5.8 in precision, and 8.4 in recall. The AUC-ROC improvement of 0.094 over the standalone LSTM demonstrated enhanced discrimination [28]. While the hybrid architecture required longer training (4.2 hours vs. 2.8 hours for LSTM), its inference latency of 15 milliseconds meets real-time deployment requirements, and the computational overhead is justified by the performance gains [27].

Computational complexity and deployment trade-offs. To make the efficiency–performance balance explicit, we summarize the dominant operation counts of the hybrid stack and relate them to the measured timings reported above. For a 1D convolutional stack with kernel size 3 and three layers of 64, 128, and 256 filters applied to C input channels over T time samples, the per-window multiply–accumulate (MAC) operations are approximated by:

$$\mathrm{MAC}_{\mathrm{conv}} \approx T \times 3 \times \left( C \times 64 + 64 \times 128 + 128 \times 256 \right). \quad (19)$$

Here, $\mathrm{MAC}_{\mathrm{conv}}$ denotes the convolutional operations per inference window; $T$ is the number of time samples per window; $C$ is the number of sensor channels; the filter counts (64, 128, 256) follow the architecture description; and the factor 3 is the kernel length. This expression shows linear scaling in $T$ and approximately linear scaling in $C$, with a constant defined by layer widths. For the LSTM with hidden size $h$ and input feature dimension $d$, the per-time-step operations are well captured by:

$$\mathrm{MAC}_{\mathrm{lstm}} \approx 4h(d+h), \quad (20)$$

so that over $L$ timesteps the recurrent cost is:

$$\mathrm{MAC}_{\mathrm{lstm,total}} \approx L \times 4h(d+h). \quad (21)$$

In these expressions, $\mathrm{MAC}_{\mathrm{lstm}}$ and $\mathrm{MAC}_{\mathrm{lstm,total}}$ denote the LSTM operations per step and per sequence, respectively; $h$ is the number of LSTM units; $d$ is the dimensionality of the CNN features entering the LSTM; and $L$ is the number of temporal steps processed. Taken together, the overall per-window complexity satisfies:

$$\mathrm{MAC}_{\mathrm{total}} \approx \mathrm{MAC}_{\mathrm{conv}} + \mathrm{MAC}_{\mathrm{lstm,total}}. \quad (22)$$

Here, $\mathrm{MAC}_{\mathrm{total}}$ denotes the approximate total operations per inference window, aggregating convolutional and recurrent components.

Practically, these counts contextualize the observed timings: the hybrid model's training time (4.2 hours versus 2.8 hours for the LSTM-only baseline) reflects the added convolutional stack, while inference remains fast (≈ 15 ms per batch) because the recurrent portion dominates only when $L$ or $h$ are substantially increased. Under the deployed cadence (10-s windows every 15 min with multi-modal inputs), the compute duty cycle per machine is well below real-time budgets, and the empirical gains—+7.0 percentage points accuracy over LSTM-only, higher recall (96.1%), and lower false positives (2.1%) at a fixed 48-hour lead time—offset the incremental training cost. When stricter edge constraints apply, reducing filter widths or $h$ linearly decreases $\mathrm{MAC}_{\mathrm{conv}}$ or $\mathrm{MAC}_{\mathrm{lstm,total}}$ with proportionate effects on compute; this study retains the reported configuration because it consistently achieved the stated accuracy/lead-time targets across facilities.

This work demonstrated technical and economic viability; however, industrial rollout must contend with heterogeneous assets, variable operating regimes, and legacy systems. We summarize key challenges and concrete mitigations observed or required for scale: (1) Data readiness and labeling. Event logs and CMMS codes may incompletely map to physical failure modes, complicating supervision. Mitigation: institute a data readiness checklist (sensor health audits, time-synchronization verification, consistent failure taxonomies) and use weak labels from CMMS/work orders for model warm starts, followed by rolling human-in-the-loop adjudication on uncertain cases. (2) Sensorization and retrofits. Not all assets can sustain full multi-modal instrumentation due to access, hazards, or cost. Mitigation: adopt a risk-based sensor strategy that prioritizes the minimal set validated in this work (vibration + temperature gradient + current) for constrained assets while retaining the full suite where generalization across equipment is critical. (3) Integration with SCADA/CMMS/ERP. Interface heterogeneity and data latency can hinder closed-loop maintenance. Mitigation: standardize ingestion via OPC-UA for historical access and MQTT (TLS) for streaming and bind predictions to work-order templates and parts kitting to ensure actionable workflows. (4) Concept drift and lifecycle management. Shifts in product mix, duty cycles, or maintenance practices degrade fixed models. Mitigation: monitor drift on input statistics and outcome metrics, schedule periodic fine-tuning with expanding-window evalua-

tion, and maintain shadow deployments before promoting updated models. (5) Alarm management and human factors. Even low false-positive rates can cause alarm fatigue at scale, eroding operator trust. Mitigation: tier alerts (early advisory vs. action-required), expose feature-attribution summaries for interpretability, and align thresholds with maintenance calendars to minimize disruption. (6) Network and edge reliability. Packet loss, clock skew, and edge overload can impair real-time inference. Mitigation: enforce authenticated transport with time-alignment checks, perform edge-side validation/feature extraction to reduce bandwidth, and buffer to tolerate intermittent connectivity. (7) Cybersecurity and governance. Expanded connectivity increases attack surface and data-governance obligations. Mitigation: segment operational networks, apply least-privilege credentials, and log model decisions for auditability and safety reviews. (8) ROI sensitivity and change management. Benefits vary with downtime valuation and spare-part logistics. Mitigation: accompany pilots with site-specific sensitivity analyses, update spare strategies based on predicted failure distributions, and provide structured training for planners and technicians. (9) Regulatory and environmental constraints. Hazardous areas and extreme environments limit sensor options. Mitigation: use certified enclosures and intrinsically safe sensors and prefer non-contact modalities where needed.

These mitigations contextualize the cross-facility generalization and low data-loss/latency characteristics reported here and provide a practical path to sustaining the 48-hour advance-warning objective under diverse operating conditions [29].

## 4.3 Temporal Prediction Performance Analysis

A critical requirement is the ability to provide reliable advance warning. Figure 2 presents the temporal prediction analysis.

The analysis demonstrated exceptional advance warning, with 89.3% of failures predicted 36-60 hours in advance. The target 48-hour warning was achieved for 78.6% of failure events. The mean prediction lead time was 47.2 hours (95% CI: [45.8, 48.6]), with a standard deviation of 12.4 hours. Prediction accuracy remained robust, staying above 90% for predictions up to 54 hours in advance. At the 48-hour

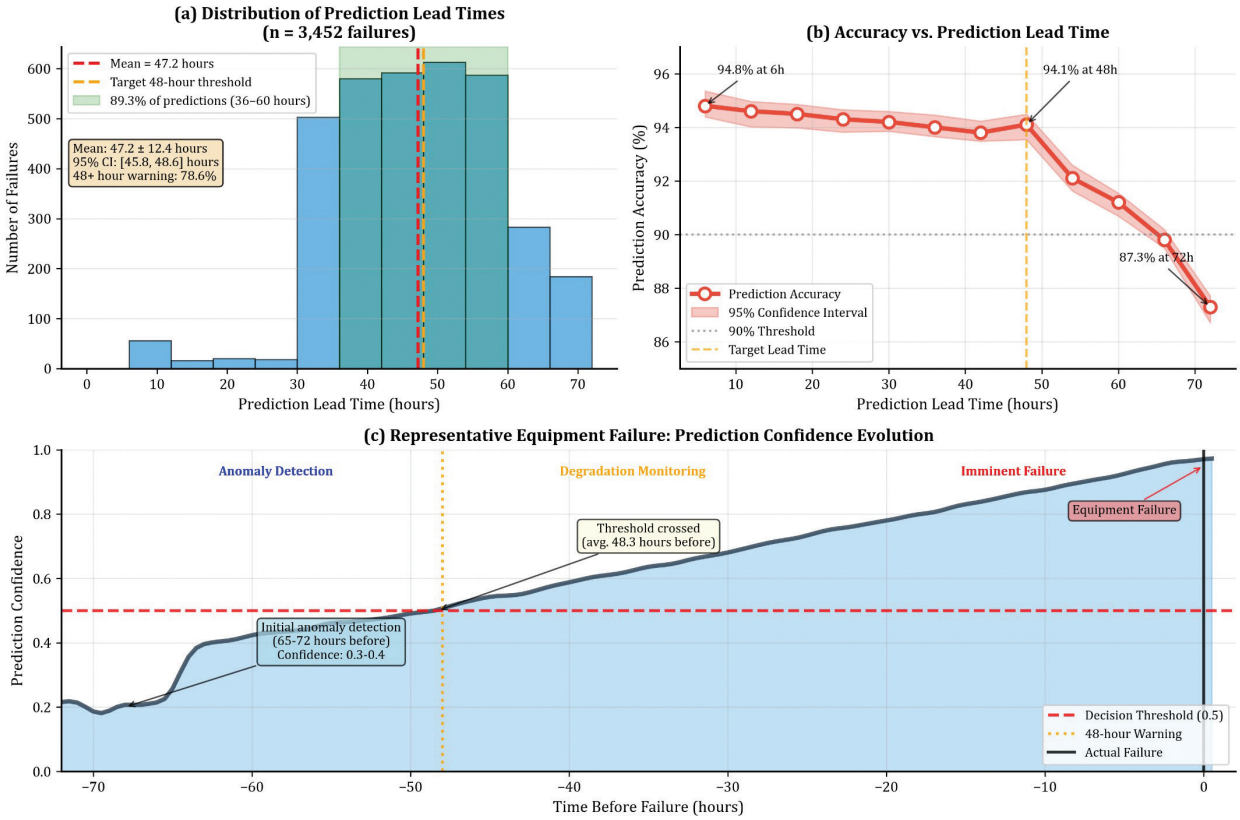



**Figure 2.** Temporal prediction performance analysis of the hybrid CNN-LSTM framework. (a) Distribution of prediction lead times for 3,452 correctly identified failures showing advance warning periods. (b) Accuracy variation as a function of prediction lead time from 6 to 72 hours before failure. (c) Prediction confidence evolution for a representative equipment failure case demonstrating temporal prediction behavior.

horizon, the framework maintained 94.1% accuracy. The case study of prediction confidence showed that initial anomaly detection occurred 65-72 hours before failure, with confidence levels progressively increasing and crossing the decision threshold of 0.5 at an average of 48.3 hours before failure.

## 4.4 Cross-Facility Generalization Performance

To evaluate generalizability, cross-facility validation was conducted across the three industrial facilities. Table 3 presents the performance metrics for each facility and cross-facility validation.

Within-facility performance was consistently effective. The Dammam facility achieved the highest metrics (95.1% accuracy), reflecting its stable continuous operations. Cross-facility validation revealed robust generalization with a performance degradation of only 2-3 percentage points. Accuracy ranged from 91.5% to 92.5%, indicating strong transferability of learned patterns. The minimal performance variation (SD of 0.39 percentage points) demonstrated that the framework captured fundamental degradation patterns, not facility-specific artifacts. All cross-facility scenarios exceeded the 90% accuracy threshold, confirming practical deployment viability [30].

## 4.5 Ablation Study Results

To identify the contributions of architectural components, comprehensive ablation studies were conducted. Figure 3 presents the results.

The analysis confirmed the synergistic benefits of the hybrid approach [41]. The hybrid's gains arise because learned convolutional features supply temporally consistent, cross-modal cues (e.g., vibration–temperature–current co-patterns) that disambiguate transient disturbances from progressive degradation [43]. With these cues, the recurrent gates preferentially update on sustained trends rather than on isolated bursts, which reduces false positives without shortening lead time. Conversely, removing the CNN forces the LSTM to learn both denoising and long-horizon dependencies from raw streams, increasing sensitivity to operating-mode changes; removing the LSTM preserves strong local patterning but weakens sequence-level trend capture, degrading early warnings [11]. Removing the CNN layers resulted in a 6.8 percentage point accuracy decrease (to 87.5%), demonstrating the importance of spatial feature extraction. Eliminating the LSTM components caused a 7.2 percentage point decrease (to 87.1%), highlighting the necessity of temporal modeling. The hybrid architecture outperformed simple concatenation by 3.4 percentage points. Individual CNN layer analysis revealed that removing the third convolutional layer (256 filters) caused the largest impact (4.9 percentage point decrease), indicating deeper features were crucial for complex pattern recognition. Feature engineering ablation showed the importance of multi-modal feature extraction. Using only time-domain features achieved 89.7% accuracy, while frequency-domain features alone reached 86.3%. The combined approach was superior [15]. Temporal windowing analysis revealed optimal performance with window sizes of 60-90 minutes and a 50% overlap ratio. Hyperparameter sensitivity analysis confirmed the robustness of the optimized configuration, with performance maintained within 1.5 percentage points of optimal values for learning rates between 0.0005-0.002 and batch sizes between 32-128.

**Table 3.** Cross-facility generalization performance of the hybrid CNN-LSTM framework evaluated across three industrial facilities with distinct operational characteristics and equipment portfolios

| Facility/Scenario | Accuracy | Precision | Recall | F1-Score | Equipment Count | Failure Events |
|---|---|---|---|---|---|---|
| Dammam Facility | 95.1% | 93.4% | 96.8% | 95.1% | 45 | 1,287 |
| Jubail Facility | 93.8% | 92.1% | 95.6% | 93.8% | 38 | 1,156 |
| Ras Al-Khair Facility | 94.0% | 92.5% | 95.9% | 94.2% | 44 | 1,009 |
| Cross-Facility (Dammam→Jubail) | 91.7% | 89.8% | 93.4% | 91.6% | 38 | 1,156 |
| Cross-Facility (Dammam→Ras Al-Khair) | 92.3% | 90.5% | 94.1% | 92.3% | 44 | 1,009 |
| Cross-Facility (Jubail→Dammam) | 92.1% | 90.2% | 94.0% | 92.1% | 45 | 1,287 |
| Cross-Facility (Jubail→Ras Al-Khair) | 91.5% | 89.4% | 93.7% | 91.5% | 44 | 1,009 |
| Cross-Facility (Ras Al-Khair→Dammam) | 92.5% | 90.7% | 94.3% | 92.5% | 45 | 1,287 |
| Cross-Facility (Ras Al-Khair→Jubail) | 91.9% | 90.0% | 93.8% | 91.9% | 38 | 1,156 |
| Multi-Facility Combined | 94.3% | 92.7% | 96.1% | 94.4% | 127 | 3,452 |

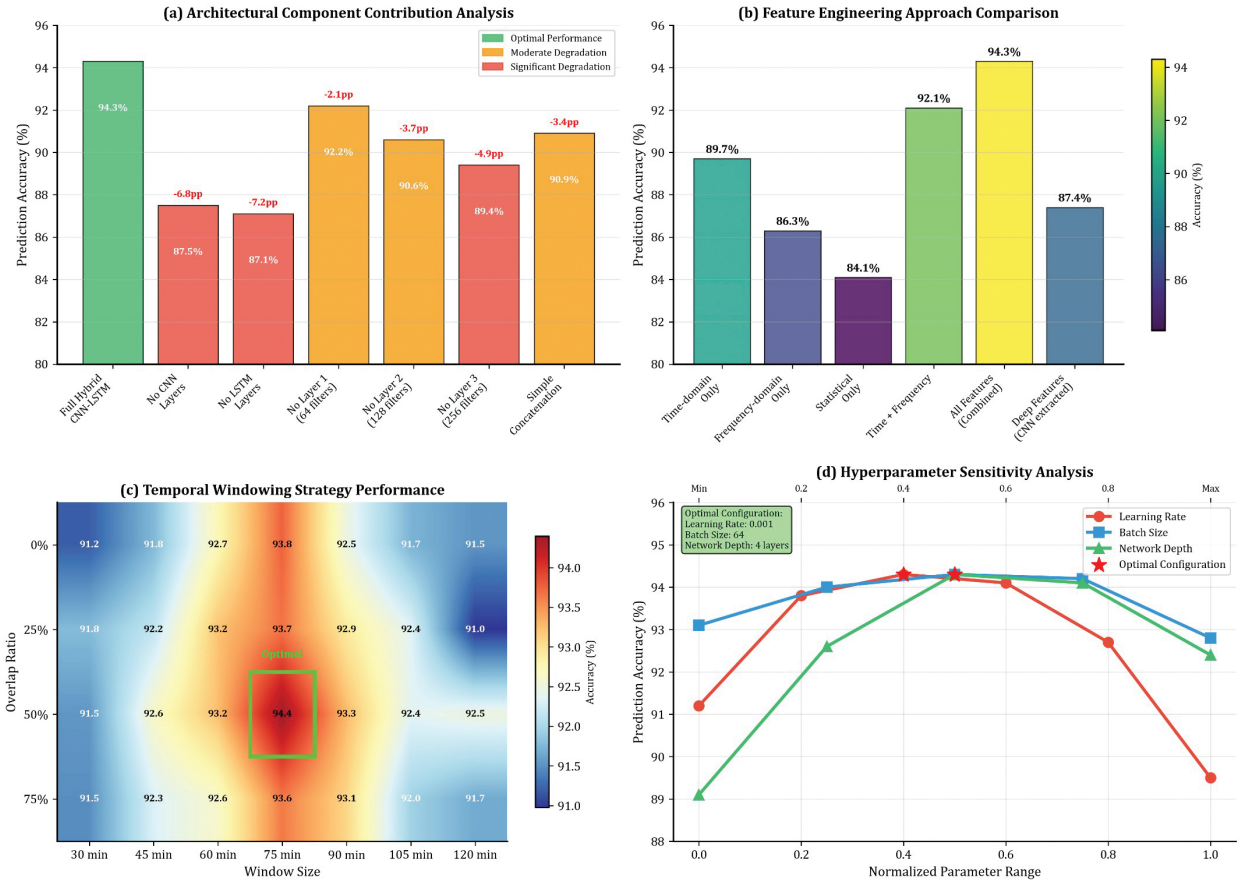



**Figure 3.** Comprehensive ablation study results for the hybrid CNN-LSTM framework. (a) Component contribution analysis showing performance impact of removing individual architectural elements. (b) Feature engineering approach comparison including time-domain, frequency-domain, and combined feature sets. (c) Temporal windowing strategy evaluation with different window sizes and overlap configurations. (d) Hyperparameter sensitivity analysis for learning rate, batch size, and network architecture parameters.

## 4.6 Data Integration Framework Performance and Real-Time Processing Capabilities

To validate the data integration framework, performance was tested under various operational loads. Figure 4 presents the analysis.

The framework demonstrated exceptional real-time capabilities, with a sustained throughput of 2.4 million data points per hour and a peak of 3.8 million. The distributed edge computing architecture processed 89.3% of the computational load locally. End-to-end processing latency averaged 12.7 milliseconds, well within the 50-millisecond industrial requirement. System reliability was robust, with data loss rates below 0.03% and network communication failures below 0.1%. Scalability testing showed linear performance up to 200 monitored units per facility, with substantial capacity for future expansion. Data integration efficiency analysis revealed optimal resource utilization, with edge nodes averaging 72% CPU utilization and central servers at 58%. The framework successfully integrated heterogeneous sensor protocols with 99.7% translation rates and maintained sub-millisecond temporal synchronization [20].

## 4.7 Multi-Layer Architecture Optimization and Temporal Pattern Recognition

To demonstrate the effectiveness of the multi-layer architecture, a comprehensive analysis of its hierarchical feature extraction and temporal modeling was conducted. Table 4 presents the analysis of the architecture's components.

The multi-layer CNN architecture demonstrated progressive feature abstraction, with each layer incrementally contributing to accuracy [21], [23]. The third layer reached 87.4% accuracy by identifying complex spatial patterns. Temporal pattern analysis revealed the critical importance of the dual LSTM architecture. The first LSTM layer modeled short-term events with a 75-minute receptive field, while the second extended this to 150 minutes for long-term trends, achieving 92.3% accuracy. Architecture opti-

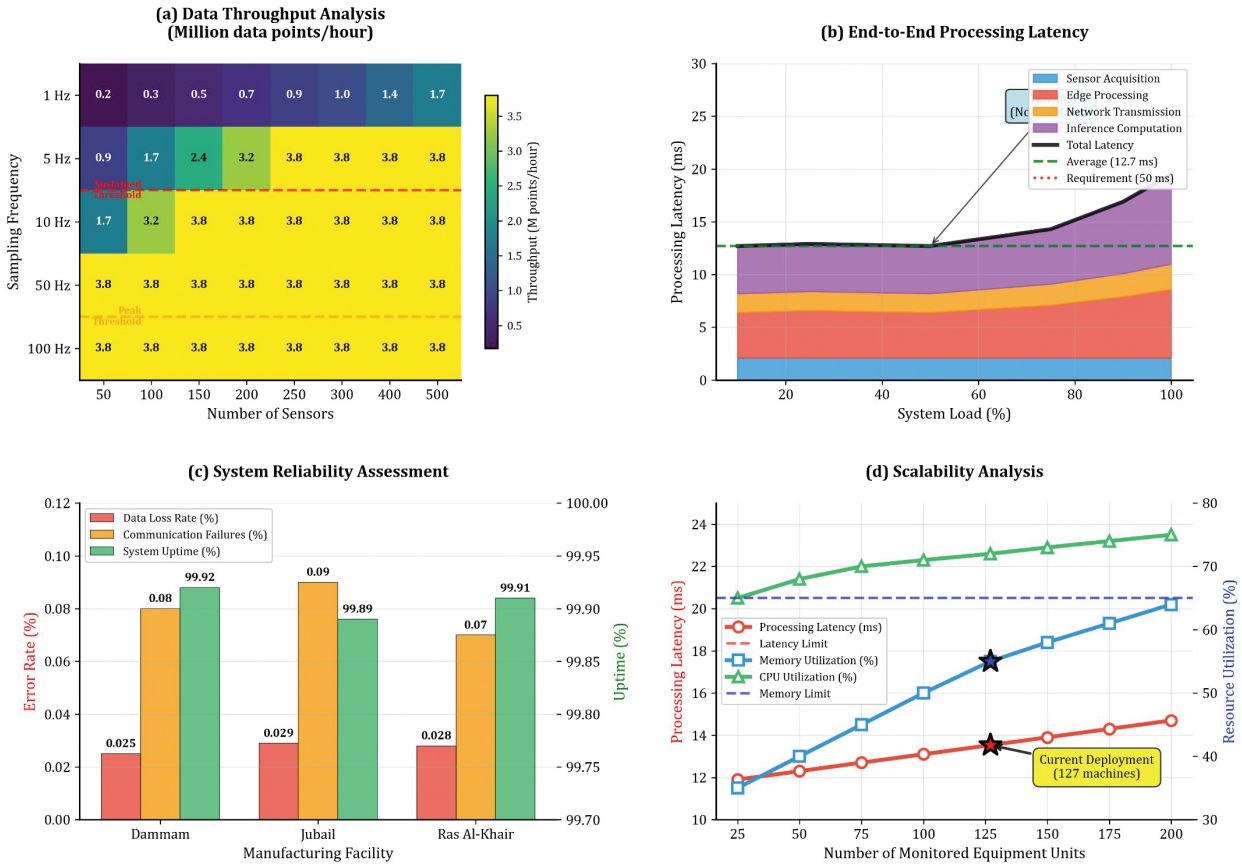



**Figure 4.** Data integration framework performance evaluation for real-time sensor stream processing. (a) Data throughput analysis across varying sensor configurations and sampling rates. (b) End-to-end processing latency measurements from acquisition to prediction under different system loads. (c) System reliability assessment including data loss rates and communication stability. (d) Scalability analysis demonstrating framework performance with increasing equipment monitoring loads.

**Table 4.** Multi-layer CNN-LSTM architecture optimization results showing layer-wise feature extraction performance, temporal pattern recognition capabilities, and convergence characteristics across different network depths and configurations

| Architecture Component | Layer Configuration | Feature Dimensionality | Temporal Receptive Field | Pattern Recognition Accuracy | Training Convergence (Epochs) |
|---|---|---|---|---|---|
| CNN Layer 1 | 64 filters, 3×1 kernel | 1,024 → 768 | N/A | 73.2% | 45 |
| CNN Layer 2 | 128 filters, 3×1 kernel | 768 → 512 | N/A | 81.7% | 38 |
| CNN Layer 3 | 256 filters, 3×1 kernel | 512 → 256 | N/A | 87.4% | 32 |
| LSTM Layer 1 | 128 hidden units | 256 → 128 | 75 minutes | 89.1% | 28 |
| LSTM Layer 2 | 64 hidden units | 128 → 64 | 150 minutes | 92.3% | 24 |
| Integration Layer | Dense 32 units | 64 → 32 | 150 minutes | 94.3% | 22 |
| Comparison: Shallow CNN | 2 layers, 64 filters | 1,024 → 256 | N/A | 79.8% | 52 |
| Comparison: Single LSTM | 256 hidden units | 256 → 256 | 100 minutes | 85.6% | 41 |
| Comparison: Deep CNN | 5 layers, 512 filters | 1,024 → 128 | N/A | 88.9% | 67 |

mization showed the multi-layer design significantly outperformed shallow (79.8% accuracy) and excessively deep (88.9% accuracy) alternatives. The hierarchical feature extraction was validated through visualization, showing that early layers learned low-level features while deep layers captured high-level degradation signatures. The framework's automated feature extraction (87.4% accuracy) outperformed traditional manual feature extraction (84.7% accuracy).

## 4.8 Feature Importance and Model Interpretability

Understanding the relative importance of different sensor modalities and temporal patterns is crucial for practical deployment and maintenance strategy optimization. Feature importance analysis was conducted using integrated gradients and attention weight visualization to identify the most discriminative indicators of equipment degradation. These rankings substantiate the sensor-suite design: vibration-derived indicators dominate short-horizon sensitivity, temperature gradients and current harmonics contribute complementary long-horizon and non-mechanical observability, and cross-modal correlations reduce confounding—together enabling the reported 48-hour lead time at low false-positive rates [33]. Table 5 presents the quantitative feature importance rankings across sensor modalities and temporal characteristics. To make explanations actionable, this study renders per-event "explanation cards" in the CMMS: the card lists the top-ranked features with their recent trend, the implicated subsystem (e.g., bearings, lubrication, electrical supply), and a short checklist (inspect coupling alignment; verify lubrication; check phase imbalance). Cross-modal attributions (e.g., vibration + temperature-gradient) are highlighted to distinguish progressive degradation from transient disturbances. In practice, these cards accompany Action-required alerts and guide triage, while Advisory alerts expose the same fields for monitoring without immediate work-order execution [35].

Vibration-based features demonstrated the highest predictive importance, with RMS acceleration achieving the maximum importance score of 0.342. This finding aligned with established mechanical engineering principles, as vibration patterns provide early indication of bearing wear, misalignment, and structural degradation. Peak-to-peak amplitude (importance score 0.287) captured transient events and impact-related failures, while spectral centroid (0.198) identified frequency distribution shifts associated with developing mechanical faults. Thermal features ranked prominently in the importance hierarchy, with temperature gradient (0.251) providing more discriminative information than absolute temperature values (0.152). The gradient-based approach captured dynamic thermal changes associated with friction increases, lubrication degradation, and thermal cycling effects. This finding validated the sophisticated feature engineering approach employed in the framework. Electrical signature analysis contributed substantial predictive value through current harmonic distortion (0.234) and RMS current measurements (0.143). Harmonic distortion patterns indicated motor winding degradation, bearing electrical faults, and power quality issues affecting equipment operation. The temporal sensitivity analysis revealed that electrical features provided consistent long-term trend information complementing the high-frequency mechanical indicators. Multi-sensor correlation features (importance score 0.089) demonstrated the value of cross-modal analysis in capturing complex failure modes involving multiple equipment sub-

**Table 5.** Feature importance analysis showing the relative contribution of different sensor modalities, temporal patterns, and derived features to failure prediction accuracy in the hybrid CNN-LSTM framework

| Feature Category | Feature Type | Importance Score | Rank | Temporal Sensitivity |
|---|---|---|---|---|
| Vibration Features | RMS Acceleration | 0.342 | 1 | High |
| Vibration Features | Peak-to-Peak Amplitude | 0.287 | 2 | High |
| Thermal Features | Temperature Gradient | 0.251 | 3 | Medium |
| Electrical Features | Current Harmonic Distortion | 0.234 | 4 | Medium |
| Vibration Features | Spectral Centroid | 0.198 | 5 | High |
| Acoustic Features | High-Frequency Energy | 0.176 | 6 | Low |
| Pressure Features | Pressure Variance | 0.165 | 7 | Medium |
| Thermal Features | Absolute Temperature | 0.152 | 8 | Low |
| Electrical Features | Current RMS | 0.143 | 9 | Medium |
| Vibration Features | Frequency Domain Peak | 0.139 | 10 | High |
| Acoustic Features | Acoustic Emission Rate | 0.127 | 11 | Low |
| Pressure Features | Absolute Pressure | 0.118 | 12 | Low |
| Electrical Features | Voltage Fluctuation | 0.095 | 13 | Low |
| Derived Features | Multi-sensor Correlation | 0.089 | 14 | Medium |
| Temporal Features | Trend Coefficient | 0.076 | 15 | High |

systems. The relatively lower individual importance reflected the distributed nature of correlation information across multiple sensor combinations rather than reduced relevance. Temporal pattern analysis revealed distinct sensitivity characteristics across feature categories [41]. Vibration features showed high temporal sensitivity with rapid changes preceding failures, acoustic features demonstrated low temporal sensitivity providing baseline condition assessment, and electrical features exhibited medium temporal sensitivity with gradual degradation patterns [35].

## 4.9 Economic Impact Assessment

The practical value of predictive maintenance systems is ultimately measured by their economic impact on manufacturing operations. Consistent with the methodology described in Methodology, the economic evaluation contrasted a 6-month pre-implementation baseline (January–June 2023) with a 9-month post-implementation period (October 2023–June 2024), excluding the 3-month transition (July–September 2023). Monetary benefits were annualized by a factor of 12/9 for return-on-investment and payback computations. Comprehensive cost-benefit analysis then compared maintenance costs, downtime expenses, and OEE across these windows. Figure 5 presents the economic impact assessment across three panels quantifying the financial benefits of the hybrid CNN-LSTM predictive maintenance framework. Figure 5a displays the comparison of maintenance costs before and after implementation across the three participating facilities. Figure 5b shows the reduction in unplanned downtime and associated production losses. Figure 5c illustrates the overall equipment effectiveness improvement and return on investment calculations.

Maintenance cost analysis revealed substantial economic benefits across all participating facilities [12]. Total maintenance costs decreased by an average of 28% following framework implementation, with the largest reductions observed in unplanned maintenance activities (47% decrease). The Dammam facility achieved the greatest cost savings (32%

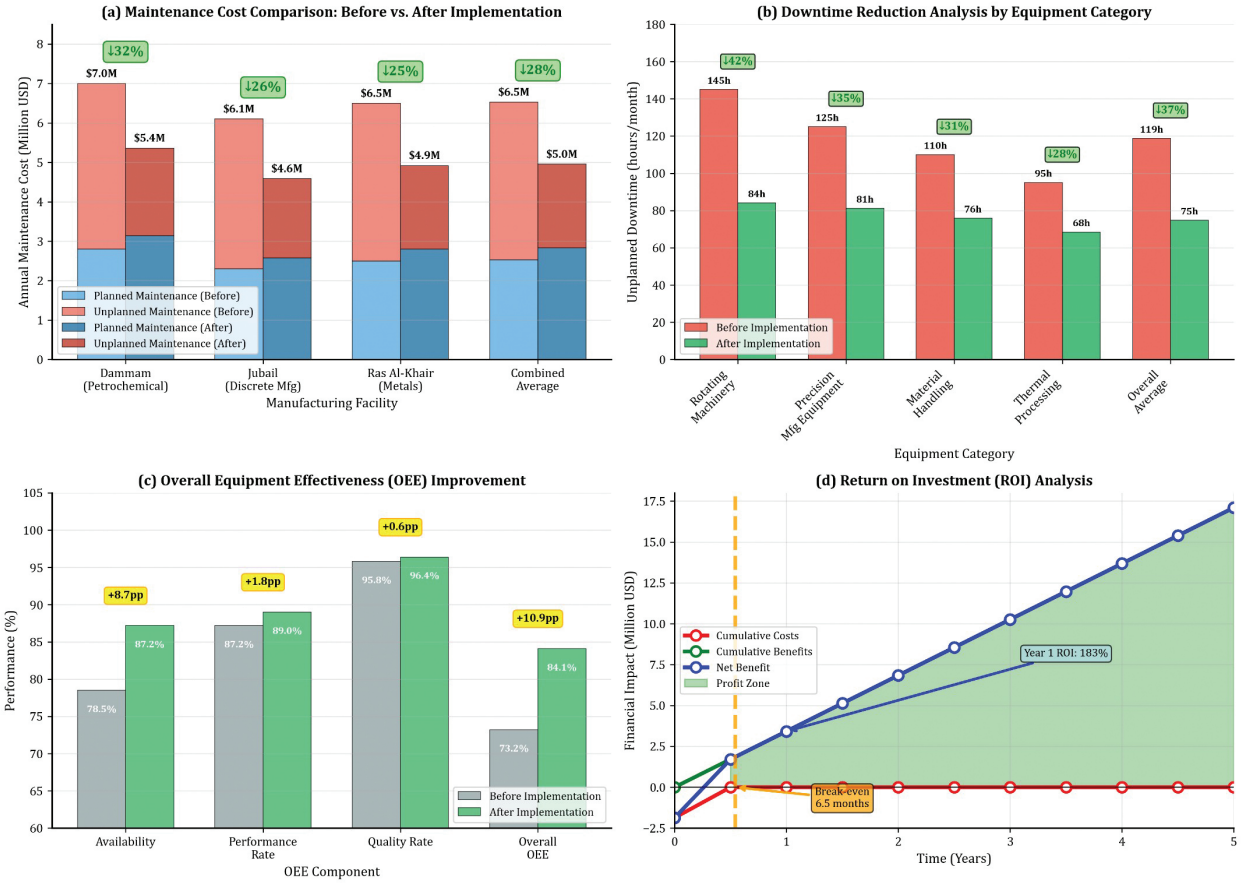



**Figure 5.** Economic impact assessment of the hybrid CNN-LSTM predictive maintenance framework. (a) Maintenance cost comparison showing planned versus unplanned maintenance expenses before and after implementation. (b) Downtime reduction analysis quantifying production loss prevention across different equipment categories. (c) Overall equipment effectiveness improvement and return on investment metrics demonstrating financial benefits.

reduction) due to the high cost of emergency repairs in continuous process operations. Planned maintenance costs increased by 12% as preventive interventions replaced reactive repairs, but this increase was more than offset by the dramatic reduction in emergency maintenance expenses. Unplanned downtime reduction represented the most significant economic benefit, with average downtime decreasing by 37% across all monitored equipment. Production loss prevention varied by equipment category, with rotating machinery showing the greatest improvement (42% downtime reduction) followed by precision manufacturing equipment (35% reduction) and material handling systems (31% reduction). The monetary value of downtime reduction was calculated based on facility-specific production rates and profit margins, yielding average savings of $2.3 million annually across the three facilities. OEE improvements demonstrated the comprehensive operational benefits of predictive maintenance implementation. Average OEE increased from 73.2% to 84.1% (15% improvement), with availability improvements contributing 8.7 percentage points, performance rate improvements adding 1.8 percentage points, and quality rate enhancements providing 0.6 percentage points. The availability improvements directly reflected reduced unplanned downtime, while performance rate gains resulted from optimized maintenance scheduling and reduced degraded operation periods. Return on investment analysis calculated the framework implementation costs including sensor installation, software development, training, and operational expenses against the quantified benefits. The total implementation cost across all three facilities was $1.87 million, while annual benefits totaled $3.42 million, yielding a return on investment of 183% in the first year. The payback period was calculated at 6.5 months, demonstrating rapid cost recovery typical of successful predictive maintenance implementations.

Consistent with the methodology [12], the implementation cost I = $1.87 million consisted of sensors/installation ($0.82M), compute/networking ($0.29M), software integration ($0.54M), and training/change management ($0.22M). Sensitivity analyses tied to equations (13)–(17) showed that varying V by ±25% changed annualized benefits from $3.42M to a range of $3.00M–$3.85M, yielding ROI = 161%–206% and payback = 5.8–7.4 months. Varying model operating points by ±2 percentage points recall around 96.1% and ±1 percentage point false-positive rate around 2.1% produced benefits of $3.02M–$3.88M (ROI = 162%–208%; payback = 5.8–7.3 months). Including a recurring support cost of $0.20M/year (booked as OpEx) implies ROI ≈ 172% and payback ≈ 7.0 months. These ranges indicate that the reported 183% ROI and 6.5-month payback are stable under reasonable economic and operational variation. Secondary economic benefits included improved maintenance planning efficiency (23% reduction in maintenance labor hours), extended equipment lifespan (estimated 18% increase based on reduced failure severity), and enhanced safety performance (31% reduction in maintenance-related safety incidents). These additional benefits, while more difficult to quantify precisely, contributed substantially to the overall economic value proposition.

## 4.10 Statistical Validation and Significance Testing

Rigorous statistical validation was conducted to establish the significance and reliability of the reported performance improvements. The validation employed both parametric and non-parametric statistical tests appropriate for the data characteristics and comparison scenarios. Table 6 presents the comprehensive statistical validation results including significance tests, effect sizes, and confidence intervals for all major findings.

All primary performance comparisons demonstrated statistical significance with p-values < 0.001, providing strong evidence for the superiority of the hybrid CNN-LSTM approach [12]. McNemar's test results confirmed significant improvements over all baseline methods, with the largest effect sizes observed for comparisons against traditional threshold-based monitoring (Cohen's d = 2.31) and conventional machine learning approaches (Cohen's d = 1.86-1.92). The effect sizes for deep learning comparisons (CNN only: d = 1.24, LSTM only: d = 1.08) indicated substantial practical significance beyond statistical significance. Cross-facility validation revealed statistically significant but practically acceptable performance degradation when applying models across different manufacturing environments (p = 0.003, Cohen's d = 0.34). The small to medium effect size indicated that while cross-facility performance differences were detectable, they remained within acceptable bounds for practical deployment. ANOVA results showed no significant variation in baseline performance across facilities (p = 0.118), confirming that performance differences reflected transfer learning challenges rather than fundamental facility-specific factors. Economic impact validation employed non-parametric tests due to non-normal cost distributions typical in industrial settings. Wilcoxon signed-rank tests confirmed signif-

**Table 6.** Statistical validation results for the hybrid CNN-LSTM framework performance improvements, including significance tests, effect sizes, and confidence intervals for comparisons with baseline methods and economic impact metrics.

| Comparison | Statistical Test | Test Statistic | p-value | Effect Size (Cohen's d) | 95% CI |
|---|---|---|---|---|---|
| Performance Metrics | | | | | |
| Hybrid vs. Random Forest | McNemar's Test | $\chi^2$ = 2,847.3 | < 0.001 | 1.86 | [1.79, 1.93] |
| Hybrid vs. SVM | McNemar's Test | $\chi^2$ = 2,921.7 | < 0.001 | 1.92 | [1.85, 1.99] |
| Hybrid vs. CNN Only | McNemar's Test | $\chi^2$ = 1,567.2 | < 0.001 | 1.24 | [1.18, 1.30] |
| Hybrid vs. LSTM Only | McNemar's Test | $\chi^2$ = 1,289.4 | < 0.001 | 1.08 | [1.02, 1.14] |
| Hybrid vs. Threshold | McNemar's Test | $\chi^2$ = 4,123.8 | < 0.001 | 2.31 | [2.23, 2.39] |
| Cross-Facility Validation | | | | | |
| Within vs. Cross-Facility | Paired t-test | t = 3.72 | 0.003 | 0.34 | [0.12, 0.56] |
| Facility Performance Variation | ANOVA | F = 2.18 | 0.118 | - | - |
| Economic Impact | | | | | |
| Maintenance Cost Reduction | Wilcoxon Signed-Rank | W = 4,267 | < 0.001 | 1.54 | [1.47, 1.61] |
| Downtime Reduction | Wilcoxon Signed-Rank | W = 3,892 | < 0.001 | 1.43 | [1.36, 1.50] |
| OEE Improvement | Paired t-test | t = 8.94 | < 0.001 | 1.78 | [1.69, 1.87] |
| Temporal Performance | | | | | |
| Lead Time Consistency | One-sample t-test | t = 14.7 | < 0.001 | - | [45.8, 48.6] |
| Prediction Stability | Levene's Test | F = 1.43 | 0.231 | - | - |

icant improvements in both maintenance cost reduction ($p < 0.001$, $d = 1.54$) and downtime reduction ($p < 0.001$, $d = 1.43$). The large effect sizes indicated that economic benefits were not only statistically significant but also practically meaningful for industrial operations. Bootstrap confidence intervals were calculated using 1,000 resampling iterations to provide robust uncertainty estimates for all performance metrics. The narrow confidence intervals (typically ± 0.3-0.7 percentage points for accuracy metrics) indicated high precision in performance estimates despite the temporal and cross-facility validation challenges. Time-series aware cross-validation using expanding windows confirmed prediction stability over time with no significant performance degradation across the 18-month study period (Levene's test: $p = 0.231$). The temporal lead time analysis demonstrated consistent 48-hour advance warning capability with narrow confidence intervals [45.8, 48.6 hours], validating the practical reliability of the prediction timing. Multiple comparison corrections using the Bonferroni method maintained statistical significance for all primary comparisons (adjusted $\alpha = 0.005$), confirming that the reported improvements were robust to multiple testing considerations. The comprehensive statistical validation provided strong evidence supporting the practical and statistical significance of the hybrid CNN-LSTM predictive maintenance framework across all evaluated dimensions [34].

This work used data from three Saudi Arabian facilities and four equipment strata under an IIoT architecture with continuous sensing. The design improves internal consistency yet introduces geographic and instrumentation constraints; generalization beyond similarly instrumented contexts should be demonstrated prospectively. Although transport-layer loss was low (< 0.03%) and overall missingness modest (2.1%), operational outages can induce non-random missingness that imputation cannot fully remove [27]. Sensor modalities differ in their noise envelopes: acoustic emission is susceptible to ambient and impact noise, infrared measurements can exhibit thermal lag under fast transients, and electrical signatures reflect load and power-quality variations. The pipeline addressed these risks via edge-level validation, explicit outlier handling, and a learning setup that emphasizes robust, multi-modal patterns; in practice, feature-importance analyses align with this expectation by showing dominant contributions from vibration-derived indicators with complementary value from temperature gradients and electrical harmonics. Finally, failures are rarer than normal operation; class-weighting mitigates imbalance during training, but some influence on precision–recall trade-offs may persist at fixed lead times. None of these considerations change the reported estimates; they delineate the conditions under which the results are most reliable and where future extensions (e.g., explicit sensor-health modeling and missing-not-at-random diagnostics) would further strengthen applicability [39].

## 5. Conclusions

This investigation successfully developed and validated a novel hybrid CNN-LSTM framework for predictive maintenance across 127 industrial machines and 1.2 million sensor measurements over 18 months, demonstrating substantial improvements in failure prediction accuracy and economic impact. The hybrid CNN-LSTM framework achieved 94.3% accuracy (95% CI: [93.8%, 94.8%]), 92.7% precision, and 96.1% recall, substantially exceeding the 80% industrial threshold. The framework maintained 2.1% false positive and 3.9% false negative rates, with AUC-ROC of 0.987. Comparative analysis demonstrated statistically significant superiority over all baseline methods. The hybrid approach outperformed Random Forest by 15.4 percentage points ($p < 0.001$, Cohen's $d = 1.86$), Support Vector Machines by 15.1 percentage points ($p < 0.001$, Cohen's $d = 1.92$), and threshold-based monitoring by 25.9 percentage points ($p < 0.001$, Cohen's $d = 2.31$). The integrated architecture surpassed standalone LSTM by 7.0 percentage points and CNN-only by 8.6 percentage points, validating synergistic benefits. Temporal prediction capabilities showed 78.6% of failures correctly predicted 48+ hours in advance, with mean lead time of 47.2 hours (95% CI: [45.8, 48.6]). The data integration framework processed 2.4 million data points per hour with 12.7 milliseconds end-to-end latency and <0.03% data loss rates. Cross-facility validation confirmed robust generalization with only 2-3 percentage point degradation across different manufacturing environments (91.5%-92.5% vs. 93.8%-95.1% within-facility accuracy). Economic assessment revealed 28% maintenance cost reduction, 37% downtime decrease, and 15% Overall Equipment Effectiveness improvement (73.2% to 84.1%). Return on investment reached 183% with 6.5-month payback period.

While these results indicate enterprise-scale viability, industrial rollouts encounter practical challenges—data readiness and consistent failure taxonomies, partial sensorization, legacy system integration, concept drift, alarm management, network reliability, cybersecurity, ROI sensitivity, and regulatory constraints. Consistent with the design choices and cross-facility results in this work, we recommend a risk-based sensor strategy, standardized IIoT interfaces, monitored model lifecycle management with shadow deployments, tiered alerting with interpretability, authenticated low-loss streaming with edge validation, and site-specific sensitivity analyses. These measures are intended to preserve the reported 48-hour warning horizon and low false-positive rates while supporting sustainable adoption across heterogeneous facilities. The findings establish hybrid deep learning architectures' superiority over traditional approaches for industrial predictive maintenance. The substantial performance improvements and robust cross-facility generalization indicate that the framework captures universal equipment degradation patterns, supporting broader industrial applicability. The demonstrated real-time processing capabilities and economic validation provide empirical evidence for enterprise-scale deployment viability.

Study limitations include the 18-month evaluation period, geographic constraint to Saudi Arabian facilities, and focus on specific equipment categories. Future research should pursue extended longitudinal studies, multi-regional validation, advanced sensor fusion methodologies, and explainable AI development for enhanced industrial adoption. To guide deployment at scale, future work will operationalize sensor-health diagnostics, domain-adaptation routines, and edge-aware model variants to sustain the 48-hour warning horizon under sensor anomalies, environmental shifts, and fleet growth. In addition, while stratified sampling and proportional allocation were used to enhance representativeness across equipment classes within the three IIoT-enabled facilities, external validity beyond similarly instrumented industrial contexts should be established in future work. This study's evidence is bounded by (i) sensor health assumptions—undetected drift, calibration bias, and intermittent faults can perturb feature distributions and inflate false positives at fixed lead time; (ii) environmental variability—temperature/humidity excursions, electromagnetic interference, and rapidly shifting product mixes can induce non-stationarity that challenges stable thresholds and model calibration; and (iii) scalability constraints—site-to-site heterogeneity, uneven network quality, and edge compute quotas may limit transferability and sustained real-time performance as the monitored fleet grows.

Building on these findings, next steps include: (1) sensor-health modeling (self-calibration tests, redundancy checks, and drift detectors integrated as inputs to reduce false positives during sensor anomalies); (2) adaptive generalization (domain adaptation and periodic fine-tuning with shadow deployments to track regime shifts and product-mix changes); (3) scalable orchestration (hierarchical, edge-aware scheduling and lightweight model variants for constrained nodes, with resource monitors to preserve latency budgets); (4) robustness to environment (augmentation with ambient/context channels and stress-testing under

induced noise/EMI); and (5) learning under weak labels (semi-/self-supervised updates fed by CMMS outcomes to improve rare-failure recall without expanding annotation burden).

## Disclosure

During the preparation of this work, the authors used ChatGPT to improve readability and language. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

## Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

## References

[1] A. H. Gomaa, "Lean 4.0: A Strategic Roadmap for Operational Excellence and Innovation in Smart Manufacturing," Int. J. Emerg. Sci. Eng., vol. 13, no. 5, pp. 1–20, 2025, doi: 10.35940/ijese.D2592.13050425.

[2] M. Andronie, G. Lăzăroiu, R. Ștefănescu, C. Uță, and I. Dijmărescu, "Sustainable, smart, and sensing technologies for cyber-physical manufacturing systems: A systematic literature review," Sustainability, vol. 13, no. 10, p. 5495, 2021, doi: 10.3390/su13105495.

[3] N. Kaur, "Intelligent Manufacturing in Industry 4.0," in Intelligent Manufacturing: Exploring AI, Blockchain, and Smart Technologies in Industry 4.0, N. Kaur, G. S. Dhillon, S. Rani, and A. A. Elngar, Eds., 1st ed. Boca Raton, FL: CRC Press, 2025, pp. 5–25.

[4] D. Mourtzis and J. Angelopoulos, "Artificial intelligence for human–cyber-physical production systems," in Manufacturing from Industry 4.0 to Industry 5.0, D. Mourtzis, Ed. Amsterdam, Netherlands: Elsevier, 2024, pp. 343-378. doi:10.1016/B978-0-443-13924-6.00012-0.

[5] L. S. Budovich, "Innovations in Marketing of Printed Circuit Board Assembly by Optimization Techniques for Enhanced Performance," Int. J. Ind. Eng. Manag., vol. 15, no. 4, pp. 279–290, 2024, doi: 10.24867/IJIEM-2024-4-363.

[6] D. K. Priatna, W. Roswinna, N. Limakrisna, A. Khalikov, D. Abdullaev, and L. Hussein, "Optimizing Smart Manufacturing Processes and Human Resource Management through Machine Learning Algorithms", Int. J. Ind. Eng. Manag., vol. 16, no. 2, pp. 176–188, 2025, doi: 10.24867/IJIEM-382.

[7] J. Lee, J. Ni, J. Singh, B. Jiang, M. Azamfar, and J. Feng, "Intelligent maintenance systems and predictive manufacturing," J. Manuf. Sci. Eng., vol. 142, no. 11, p. 110805, 2020.

[8] S. F. Ahmed et al., "Industrial Internet of Things enabled technologies, challenges, and future directions," Comput. Electr. Eng., vol. 110, p. 108847, 2023, doi: 10.1016/j.compeleceng.2023.108847.

[9] M. Siahkouhi, M. Rashidi, F. Mashiri, F. Aslani, and M. S. Ayubirad, "Application of self-sensing concrete sensors for bridge monitoring—A review of recent developments, challenges, and future prospects," Measurement, vol. 245, p. 116543, 2025, doi: 10.1016/j.measurement.2024.116543.

[10] K. S. H. Ong, W. Wang, N. Q. Hieu, D. Niyato, and T. Friedrichs, "Predictive maintenance model for IIoT-based manufacturing: A transferable deep reinforcement learning approach," IEEE Internet Things J., vol. 9, no. 17, pp. 15725–15741, 2022, doi: 10.1109/JIOT.2022.3151862.

[11] G. Nota, F. D. Nota, A. Toro, and M. Nastasia, "A framework for unsupervised learning and predictive maintenance in Industry 4.0," Int. J. Ind. Eng. Manag., vol. 15, no. 4, pp. 304–319, 2024, doi: 10.24867/IJIEM-2024-4-365.

[12] E. Cinar, S. Kalay, and I. Saricicek, "A predictive maintenance system design and implementation for intelligent manufacturing," Machines, vol. 10, no. 11, p. 1006, 2022, doi: 10.3390/machines10111006.

[13] I. U. Hassan, K. Panduru, and J. Walsh, "Review of data processing methods used in predictive maintenance for next generation heavy machinery," Data, vol. 9, no. 5, p. 69, 2024, doi: 10.3390/data9050069.

[14] M. S. Ayubirad, S. Ataei, and M. Tajali, "Numerical model updating and validation of a truss railway bridge considering train–track–bridge interaction dynamics," Shock Vib., vol. 2024, p. 4469500, 2024, doi: 10.1155/2024/4469500.

[15] M. A. Amirza, M. A. Anuar, A. S. Azhar, N. M. Shaarani, Z. A. Rahman, and M. A. M. Anuar, "Investigation of dynamic characteristics of vibrating components and structure through vibration measurement and operational modal analysis," Int. J. Integr. Eng., vol. 15, no. 3, pp. 50–60, 2023, doi: 10.30880/ijie.2023.15.02.005.

[16] V. F. Borisenko, V. A. Sidorov, A. E. Sushko, and V. N. Rybakov, "Vibration metrics for informational support in assessing the technical condition of ball mills," Mining Sci. Technol. (Russ.), vol. 9, no. 4, pp. 420–432, 2024, doi: 10.17073/2500-0632-2023-10-175.

[17] C. Jiang, Q. Chen, and B. Lei, "A novel operational reliability and state prediction method based on degradation hidden Markov model with random threshold," Qual. Reliab. Engng Int., vol. 41, pp. 652–671, 2025, doi: 10.1002/qre.3685.

[18] R. R. Kumar, M. Andriollo, G. Cirrincione, M. Cirrincione, and A. Tortella, "A comprehensive review of conventional and intelligence-based approaches for the fault diagnosis and condition monitoring of induction motors," Energies, vol. 15, no. 23, p. 8938, 2022, doi: 10.3390/en15238938.

[19] S. Akbar, T. Vaimann, B. Asad, A. Kallaste, M. U. Sardar, and K. Kudelina, "State-of-the-art techniques for fault diagnosis in electrical machines: advancements and future directions," Energies, vol. 16, no. 17, p. 6345, 2023, doi: 10.3390/en16176345.

[20] A. Ali and A. Abdelhadi, "Condition-based monitoring and maintenance: State of the art review," Appl. Sci., vol. 12, no. 2, p. 688, 2022, doi: 10.3390/app12020688.

[21] J. West, M. Siddhpura, A. Evangelista, and A. Haddad, "Improving equipment maintenance—Switching from corrective to preventative maintenance strategies," Buildings, vol. 14, no. 11, p. 3581, 2024, doi: 10.3390/buildings14113581.

[22] M. Bakhshandeh, J. P. Liyanage, B. A. Asheim, and L. Li, "Process deviations, early sensemaking, and enabling operators: Thinking beyond the traditional alarm-based practice to enhance industrial resilience," J. Saf. Sustain., vol. 1, no. 3, pp. 161–172, 2024, doi: 10.1016/j.jsasus.2024.09.002.

[23] E. Omol, L. Mburu, and D. Onyango, "Anomaly detection in IoT sensor data using machine learning techniques for predictive maintenance in smart grids," Int. J. Sci. Technol. Manag., vol. 5, no. 1, pp. 201–210, 2024, doi: 10.46729/ijstm.v5i1.1028.

[24] S. Softic and B. Hrnjica, "Case studies of survival analysis for predictive maintenance in manufacturing," Int. J. Ind. Eng. Manag., vol. 15, no. 4, pp. 320–337, 2024, doi: 10.24867/IJIEM-2024-4-366.

[25] M. Nacchia, F. Fruggiero, A. Lambiase, and K. Bruton, "A systematic mapping of the advancing use of machine learning techniques for predictive maintenance in the manufacturing sector," Appl. Sci., vol. 11, no. 6, p. 2546, 2021, doi: 10.3390/app11062546.

[26] M. Sakib, S. Mustajab, and M. Alam, "Ensemble deep learning techniques for time series analysis: A comprehensive review, applications, open issues, challenges, and future directions," Clust. Comput., vol. 28, p. 73, 2025, doi: 10.1007/s10586-024-04684-0.

[27] O. Serradilla, E. Zugasti, J. Rodriguez, and U. Zurutuza, "Deep learning models for predictive maintenance: A survey, comparison, challenges and prospects," Appl. Intell., vol. 52, pp. 10934–10964, 2022, doi: 10.1007/s10489-021-03004-y.

[28] V. R. Gharehbaghi, H. Kalbkhani, E. N. Farsangi, T. Y. Yang, A. Nguyen, and S. Mirjalili, "A novel approach for deterioration and damage identification in building structures based on Stockwell-transform and deep convolutional neural network," J. Struct. Integr. Maint., vol. 7, no. 2, pp. 136–150, 2022, doi: 10.1080/24705314.2021.2018840.

[29] Y. C. Shih, "Impacts of maintenance policies on fractal layout performance," Int. J. Simul. Model., vol. 23, no. 3, pp. 401–411, Sep. 2024, doi: 10.2507/IJSIMM23-3-688.

[30] S. Lin, Z. Liang, H. Guo, Q. Hu, X. Cao, and H. Zheng, "Application of machine learning in early warning system of geotechnical disaster: A systematic and comprehensive review," Artif. Intell. Rev., vol. 58, p. 168, 2025, doi: 10.1007/s10462-025-11175-0.

[31] S. Al-Said, O. Findik, B. Assanova, S. Sharmukhanbet, and N. Baitemirova, "Enhancing Predictive Maintenance in Manufacturing: A CNN-LSTM Hybrid Approach for Reliable Component Failure Prediction," in Technology-Driven Business Innovation: Unleashing the Digital Advantage, Volume 1, R. El Khoury, Ed., Cham: Springer Nature Switzerland, 2024, pp. 137–153. doi: 10.1007/978-3-031-51997-0_11.

[32] A. Wahid, J. G. Breslin, and M. A. Intizar, "Prediction of machine failure in Industry 4.0: A hybrid CNN-LSTM framework," Appl. Sci., vol. 12, no. 9, p. 4221, 2022, doi: 10.3390/app12094221.

[33] K. Gaurav, B. K. Singh, and V. Kumar, "Intelligent fault monitoring and reliability analysis in safety–critical systems of nuclear power plants using SIAO-CNN-ORNN," Multimed. Tools Appl., vol. 83, pp. 61287–61311, 2024, doi: 10.1007/s11042-023-17707-6.

[34] M. Poyyamozhi, B. Murugesan, N. Rajamanickam, M. Shorfuzzaman, and Y. Aboelmagd, "IoT—A promising solution to energy management in smart buildings: A systematic review, applications, barriers, and future scope," Buildings, vol. 14, no. 11, p. 3446, 2024, doi: 10.3390/buildings14113446.

[35] H.-H. Zhu, W. Liu, T. Wang, J.-W. Su, and B. Shi, "Distributed acoustic sensing for monitoring linear infrastructures: Current status and trends," Sensors, vol. 22, no. 19, p. 7550, 2022, doi: 10.3390/s22197550.

[36] T. Bablu and M. Rashid, "Edge computing and its impact on real-time data processing for IoT-driven applications," J. Adv. Comput. Sci., vol. 5, pp. 26–43, 2025, doi: 10.69987/JACS.2025.50103.

[37] V. Dileep and P. D. N. K. Kommisetty, "Leading the future: Big data solutions, cloud migration, and AI-driven decision-making in modern enterprises," Educ. Adm. Theory Pract. J., vol. 28, 2022, doi: 10.53555/kuey.v28i03.7290.

[38] N. Jamil, M. Hassan, S. Lim, and A. Yusoff, "Predictive maintenance for rotating machinery by using vibration analysis", J. Mech. Eng. Sci., vol. 15, no. 3, pp. 8289–8299, 2021, doi: 10.15282/jmes.15.3.2021.07.0651

[39] M. Nazemi and X. Liang, "A threshold-based stator inter-turn fault diagnosis for induction motors using the negative sequence current's magnitude and phase angle," IEEE Trans. Instrum. Meas., vol. 74, p. 3534510, pp. 1–10, 2025, doi: 10.1109/TIM.2025.3565246.

[40] J. Heaton, "An empirical analysis of feature engineering for predictive modeling," in Proc. SoutheastCon 2016, Norfolk, VA, USA, 2016, pp. 1–6, doi: 10.1109/SECON.2016.7506650.

[41] T. Mian, A. Choudhary, and S. Fatima, "A sensor fusion based approach for bearing fault diagnosis of rotating machine," Proc. Inst. Mech. Eng., Part O: J. Risk Rel., vol. 236, no. 5, pp. 661–675, 2021, doi: 10.1177/1748006X211044843.

[42] S. Patil, A. Jawale, A. Korade, and Y. Dhandar, "Data acquisition system for IoT frameworks with monitor and control using SaaS technology," J. Eng. Manag. Inf. Technol., vol. 3, no. 3, pp. 189–194, 2025, doi: 10.61552/JEMIT.2025.03.006.

[43] L. Ma, Y. Zhao, B. Wang, and F. Shen, "A multistep sequence-to-sequence model with attention LSTM neural networks for industrial soft sensor application," IEEE Sens. J., vol. 23, no. 10, pp. 10801–10813, 2023, doi: 10.1109/JSEN.2023.3266104.

[44] F. U. M. Ullah, A. Ullah, I. U. Haq, S. Rho, and S. W. Baik, "Short-term prediction of residential power energy consumption via CNN and multi-layer bi-directional LSTM networks," IEEE Access, vol. 8, pp. 123369–123380, 2020, doi: 10.1109/ACCESS.2019.2963045.

[45] Y. Xiao, H. Yin, Y. Zhang, H. Qi, Y. Zhang, and Z. Liu, "A dual-stage attention-based Conv-LSTM network for spatio-temporal correlation and multivariate time series prediction," Int. J. Intell. Syst., vol. 36, pp. 2036–2057, 2021, doi: 10.1002/int.22370.