

GAZI

## JOURNAL OF ENGINEERING SCIENCES

# Hybrid Deep Learning-Based Fault Prediction for Industrial Machinery Using CNN-LSTM and CNN-GRU: CLKKSM Facility Application and Data-Driven Predictive Maintenance Action Plan

Salih Ermiş<sup>a</sup>, Oğuz Taşdemir<sup>b</sup>, Çağatay Başgöl<sup>c</sup>

Submitted: 04.05.2026 Revised: 09.06.2026 Accepted: 16.06.2026 doi:10.30855/gmbd.070526A01

<sup>a</sup> Kırşehir Ahi Evran University, Faculty of Engineering and Architecture, Dept. of Electrical and Electronics Engineering 40100 -Kırşehir, Turkey, Orcid: 0000-0002-1053-9160

<sup>b</sup> Kırşehir Ahi Evran University, Faculty of Engineering and Architecture, Dept. of Electrical and Electronics Engineering 40100 -Kırşehir, Turkey, Orcid: 0000-0003-1782-0024

<sup>c</sup> Kırşehir Ahi Evran University, Institute of Science, Dept. of Electrical and Electronics Engineering 40100 -Kırşehir, Turkey, Orcid: 0009-0009-2198-9122

\*Corresponding author: sermis@ahievran.edu.tr

## ABSTRACT

**Keywords:** CNN-LSTM, CNN-GRU, fault prediction, predictive maintenance, time series forecasting.

Unplanned machine downtime in manufacturing facilities leads to significant production losses, rising maintenance costs, and occupational safety risks. This study compares two hybrid deep learning architectures—Convolutional Neural Network–Long Short-Term Memory (CNN-LSTM) and Convolutional Neural Network–Gated Recurrent Unit (CNN-GRU) —for monthly fault frequency prediction using an eight-year operational dataset from the CLKKSM facility. One-dimensional convolutional layers extract local temporal features from the raw time series, which are subsequently encoded by recurrent cells to capture long-range dependencies. Both models were trained on historical data and validated on a held-out period, then evaluated against actual fault events in early 2026. CNN-LSTM consistently outperformed CNN-GRU across all error metrics on the out-of-sample test set. The performance gap is attributed to the four-gate memory mechanism of LSTM, which models medium- and long-range temporal dependencies more effectively than the two-gate GRU structure. Building on CNN-LSTM forecasts, a three-phase predictive maintenance action plan was developed for the April–December 2026 period, incorporating machine-level priority classification, risk-stratified schedules, and trackable Key Performance Indicator (KPI) targets. The proposed methodology bridges the gap between model outputs and field implementation, offering a transferable framework for data-driven predictive maintenance in industrial settings.

## 1. Introduction

Unplanned machine downtime in manufacturing facilities directly affects not only output loss but also maintenance costs and occupational safety risks. As Jardine et al. [1] define, condition-based maintenance (CBM) is "a maintenance program that recommends maintenance decisions based on the information collected through condition monitoring" and "consists of three main steps: data acquisition, data processing and maintenance decision-making"; however, conventional time-based maintenance programmes operate independently of the machine's current condition and may therefore give rise to

**To cite this article:** S. Ermiş, O. Taşdemir and Ç. Başgöl, "Hybrid Deep Learning-Based Fault Prediction for Industrial Machinery Using CNN-LSTM and CNN-GRU: CLKKSM Facility Application and Data-Driven Predictive Maintenance Action Plan", *Gazi Journal of Engineering Sciences*, vol. 12, no. 2, pp. 241-254, 2026. doi:10.30855/gmbd.070526A01

both unnecessary interventions and faults that go undetected in time. Data-driven methods aimed at identifying fault patterns in advance have emerged as the most promising route out of this dilemma [2, 3].

Deep learning architectures—particularly Long Short-Term Memory (LSTM) [4] and Gated Recurrent Unit (GRU) [5] cells, which stand out for their capacity to learn time-series patterns—have found an increasingly broad application base in the field of fault prediction in recent years. Convolutional Neural Network–Recurrent Neural Network (CNN-RNN) hybrid models that combine the local feature-extraction power of one-dimensional convolution operations with the contextual memory capability of recurrent networks yield additional gains in prediction performance [6, 7].

This study was designed around two primary objectives: (i) to experimentally compare CNN-LSTM and CNN-GRU models on the eight-year fault dataset of the CLKKSM facility and to validate them against actual events in the first quarter of 2026; and (ii) to translate the predictions of the better-performing model into a concrete maintenance implementation schedule covering the April–December 2026 period.

The remainder of the paper proceeds as follows: Section 2 summarises related prior work, Section 3 introduces the dataset and preprocessing steps, Section 4 describes the model architectures and training setup, Section 5 presents the experimental findings, Section 6 discusses the results, Section 7 develops the predictive maintenance action plan, and Section 8 provides the general evaluation and future research directions.

## 2. Related Works

### 2.1. Condition-based and predictive maintenance frameworks

Condition-based maintenance (CBM) strategies were given a theoretical foundation through the systematic framework of Jardine et al. [1]. Lee et al. [2] highlighted that progress requires "a transformation of maintenance strategy from the traditional fail-and-fix practices (diagnostics) to a predict-and-prevent methodology (prognostics)"; this work continues to be one of the important reference sources in today's predictive maintenance (PdM) literature.

On the time-series forecasting front, Hochreiter and Schmidhuber [4] proposed LSTM to resolve the vanishing gradient problem through gating mechanisms. The GRU architecture [5] targeted a similar long-term dependency capacity using fewer parameters. LeCun et al. [6] demonstrated that deep learning allows computational models "to learn representations of data with multiple levels of abstraction."

Hybrid CNN-LSTM architectures have reported concrete performance gains in vibration-signal-based machine health monitoring: Zhao et al. [7] designed "a deep neural network structure named Convolutional Bi-directional Long Short-Term Memory networks (CBLSTM)" and showed that "The experimental results have shown that our model is able to outperform several state-of-the-art baseline methods." In the domain of remaining useful life (RUL) estimation, Wu et al. [8] and Li et al. [9] established the importance of deep learning approaches for RUL estimation as a challenging but critical prognostics problem. Lee et al. [2] further state that "The goals of Prognostics and Health Management (PHM) include maximizing the operational availability, reduction of maintenance costs and improvement of system reliability and safety." To the best of the authors' knowledge, there is a limited number of integrated studies predictions generated by machine learning models are directly transferred to the field. This study aims to close that gap.

### 2.2. Deep learning architectures: from recurrent networks to attention mechanisms

The evolution of deep learning architectures underpinning modern fault prediction systems can be traced along two parallel lines: recurrent sequence modellers and attention-based transformers. On the recurrent side, the foundational contributions of Hochreiter and Schmidhuber [4] and Cho et al. [5] established LSTM and GRU cells as the dominant tools for capturing temporal dependencies in industrial time-series data. The theoretical basis for combining these recurrent cells with convolutional feature extractors was consolidated by LeCun et al. [6], whose demonstration that deep models learn hierarchical data representations motivated the CNN-RNN hybrid paradigm explored in this study.

A decisive architectural shift occurred with the introduction of the Transformer by Vaswani et al. [10], who proposed a

network architecture based entirely on self-attention mechanisms, dispensing with recurrence and convolution. The multi-head self-attention formulation enables the model to jointly attend to information from different representation subspaces at different sequence positions, theoretically providing an advantage over LSTM in capturing very long-range dependencies. However, the permutation-invariant nature of the self-attention mechanism raises a non-trivial concern for univariate industrial time series: Zeng et al. [11] demonstrated that Transformer-based long-term forecasting models can be outperformed by simple linear baselines on several real-world datasets, attributing this finding to temporal information loss introduced by positional-encoding schemes. This empirical result provides an important motivation for the present study's choice of CNN-LSTM and CNN-GRU architectures over Transformer-based alternatives for short univariate monthly fault-count sequences; it also defines a clear benchmark for future architectural comparisons.

Within the specific domain of machine fault detection and diagnosis, Lei et al. [12] provided a comprehensive review and roadmap of machine learning applications, identifying data-driven prognostics as the most active research frontier. Their survey established that the transition from hand-crafted feature engineering to end-to-end deep representation learning constitutes the dominant trend across bearing, gear, and rotating machinery diagnostics. Complementing this broad perspective, Zhao et al. [13] conducted an in-depth survey of deep learning applications to machine health monitoring, reviewing auto-encoder variants, Restricted Boltzmann Machines, CNNs, and RNNs; their analysis confirmed that hybrid architectures combining convolutional and recurrent processing consistently outperform single-paradigm models on raw sensor data.

At the intersection of attention mechanisms and machine health monitoring, Zhao et al. [14] proposed local feature-based Gated Recurrent Unit (GRU) networks that combine handcrafted feature design with automatic feature learning, demonstrating that hybrid feature extraction strategies consistently outperform single-paradigm deep learning models; this finding directly supports the architectural choices made in the present study. Woo et al. [15] further formalised the concept of channel and spatial attention with the Convolutional Block Attention Module (CBAM), a lightweight plug-in that applies sequential attention to convolutional feature maps; its low parameter overhead makes it a promising candidate for augmenting the CNN feature-extraction stage of the hybrid architectures evaluated here.

On the maintenance scheduling side, Yu et al. [16] proposed an integrated dynamic predictive maintenance framework combining a deep auto-encoder for feature extraction with a deep forest for failure probability estimation, spanning the complete pipeline from raw sensor input to maintenance decision. Their work demonstrated that tightly coupling prognostic models with scheduling logic yields measurable reductions in maintenance cost and unplanned downtime, a finding that directly motivates the action-plan component developed in Section 7 of the present study.

Finally, the challenge of limited training data has been addressed in the broader time-series literature through data augmentation strategies. Wen et al. [17] surveyed augmentation techniques for deep learning on time-series data, identifying window slicing, magnitude warping, and synthetic minority oversampling as the most effective methods for small-sample industrial datasets. Incorporating such techniques in future retraining cycles, alongside multivariate sensor inputs and Transformer-based architectures, constitutes the primary direction for extending the present methodology beyond its current univariate, 96-month training horizon.

### 3. Dataset and Preprocessing

#### 3.1. Data source and descriptive analysis

The fault records of five machines—CLKKS1 through CLKKS5—operating at the CLKKSM facility between January 2018 and December 2025 constitute the training data of the study. In addition, the January–March 2026 period was set aside as an independent test set for model validation. Each record contains four attributes: machine identifier, event date-time, fault category (Mechanical or Electrical), and total downtime requiring intervention (in minutes). During the data-cleaning phase, rows containing negative downtime values were removed from the set. Table 1 presents summary statistics for both sets.

Table 1. Dataset descriptive statistics

| Parameter           | Training (2018–2025)   | Test (2026)            | Description               |
|---------------------|------------------------|------------------------|---------------------------|
| Number of Records   | 5,748                  | 234                    | Excluding invalid records |
| Number of Machines  | 5 (CLKKS1–5)           | 5 (CLKKS1–5)           | Identical hardware        |
| Fault Category      | Mechanical, Electrical | Mechanical, Electrical | Binary classification     |
| Monthly Avg. Faults | ≈60.1 events           | ≈78 events             | Based on monthly totals   |
| Avg. Downtime       | 77 min/event           | 54 min/event           | Positive records only     |

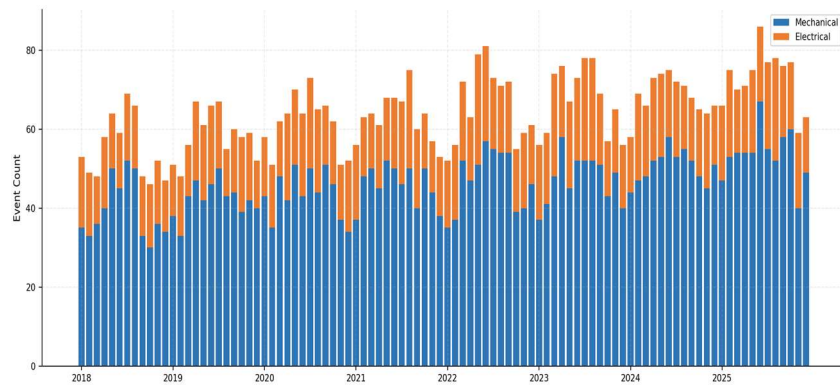
Raw daily records were aggregated to the monthly level, computing for each month the total event count, cumulative downtime, and average downtime. Figure 1 visually conveys the three main time-series dimensions for the 2018–2025 period.



(a)



(b)



(c)

Figure 1. (a) Monthly fault count, (b) Monthly cumulative downtime, (c) Fault category distribution by month (2018–2025)

## 4. Methodology

### 4.1. CNN feature extraction layer

A one-dimensional convolutional neural network (1-D CNN) preprocessing module was designed to uncover short-term patterns embedded in the raw fault-count time series. This module contains  $n = 8$  independent convolution filters of length  $k = 3$ . Each filter is slid across the input window; a Rectified Linear Unit (ReLU) non-linearity is applied to the output, and then global max-pooling yields a single scalar representing the entire window. As a result, a time slice of  $WIN = 12$  months is transformed as formulated in Eq.1. and formulated into an 8-dimensional feature vector to be passed to the downstream recurrent cell:

$$h_{conv(t)} = ReLU(\sum_k W_k \cdot x_{\{t-k\}} + b) \quad (1)$$

where  $W_k$  denotes the filter weights and  $b$  the bias term.

### 4.2. Gating mechanisms: LSTM and GRU

After receiving the CNN feature vector, two different recurrent cell structures were evaluated independently of each other.

In the CNN-LSTM model, the four-gate chain is defined by Eqs. (2) – (6):

$$f_t = \sigma(W_f \cdot [h_{\{t-1\}}, x_t] + b_f) \text{ (Forget gate)} \quad (2)$$

$$i_t = \sigma(W_i \cdot [h_{\{t-1\}}, x_t] + b_i) \text{ (Input gate)} \quad (3)$$

$$o_t = \sigma(W_o \cdot [h_{\{t-1\}}, x_t] + b_o) \text{ (Output gate)} \quad (4)$$

$$c_t = f_t \odot c_{\{t-1\}} + i_t \odot \tanh(W_c \cdot [h_{\{t-1\}}, x_t] + b_c) \text{ (Cell memory)} \quad (5)$$

$$h_t = o_t \odot \tanh(c_t) \text{ (Hidden state output)} \quad (6)$$

In the CNN-GRU model, a two-gate structure with fewer parameters is used, as defined by Eqs. (7)–(9):

$$z_t = \sigma(W_z \cdot [h_{\{t-1\}}, x_t] + b_z) \text{ (Update gate)} \quad (7)$$

$$r_t = \sigma(W_r \cdot [h_{\{t-1\}}, x_t] + b_r) \text{ (Reset gate)} \quad (8)$$

$$h_t = (1 - z_t) \odot h_{\{t-1\}} + z_t \odot \tanh(W_n \cdot [r_t \odot h_{\{t-1\}}, x_t] + b_n) \text{ (New hidden state)} \quad (9)$$

In the equations,  $\sigma$  denotes the sigmoid function,  $\odot$  element-wise (Hadamard) multiplication,  $W_*$  the learned weight matrices, and  $b_*$  the bias vectors.

### 4.3. Model training and hyperparameters

Both models were trained with hidden dimension  $h = 32$ , mini-batch size 8, and a total of 500 epochs. Gradient updates were performed with Adam optimisation [18] configured with a step size of 0.004,  $\beta_1 = 0.9$  and  $\beta_2 = 0.999$ . Input fault counts were scaled to the  $[0.05, 0.95]$  range via Min-Max normalisation. The last 12 months of observations—corresponding to 2025—were held out as a validation set, leaving 72 monthly observations for training. The 2026 forecast was generated with a one-step autoregressive strategy: the predicted value for each month was inserted into the input window for the next step, completing the full twelve-month horizon. As a characteristic of autoregressive forecasting with a mean-reverting model, the CNN-LSTM predictions stabilise at a fixed point (124.5 events/month) from October onward; this reflects the model converging to its learned long-run mean and should be interpreted as an upper-bound plateau rather than a precise monthly estimate for the late-2026 period [19]. Notably, this plateau value of 124.5

events/month independently satisfies the Critical risk criterion ( $\geq 105$  events/month) defined in the maintenance action plan (Section 7.1), providing an additional methodological justification for the CRITICAL designation assigned to October–December 2026

## 5. Experimental Results

### 5.1. Training behaviour and validation performance

Figure 2 summarises the training process and validation-set performance of both models. The CNN-LSTM model achieved Mean Absolute Error (MAE) = 14.00 and Mean Absolute Percentage Error (MAPE) = 26.4% on the 2025 validation set. The model produced systematic over-predictions of 10–25 events above actual values throughout 2025. The model fell below actual values in only two months: August 2025 (actual = 76, predicted = 68.4, error =  $-7.6$ ) and December 2025 (actual = 67, predicted = 61.0, error =  $-6.0$ ).

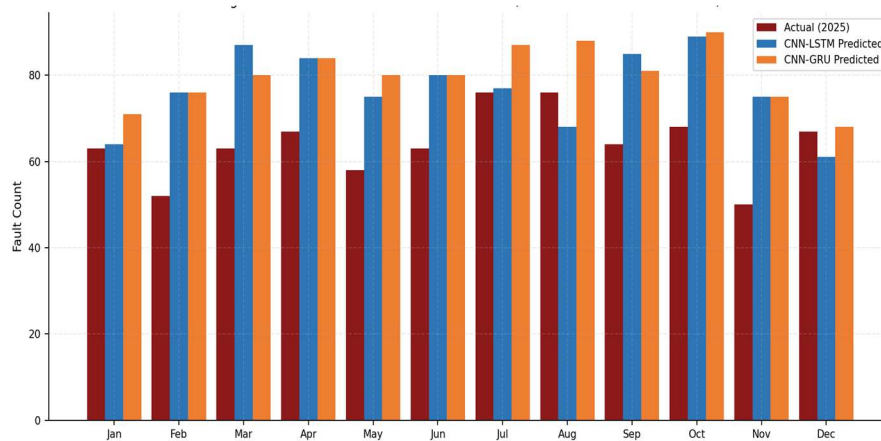


Figure 2. Comparison of the predicted values from the CNN-LSTM and CNN-GRU models with the actual values

### 5.2. 2026 Monthly forecasts

Figure 3 present the twelve-month forecast results obtained by the autoregressive method together with the January–March 2026 actuals. Table 2 shows the monthly forecast values alongside their corresponding risk class and recommended maintenance mode.

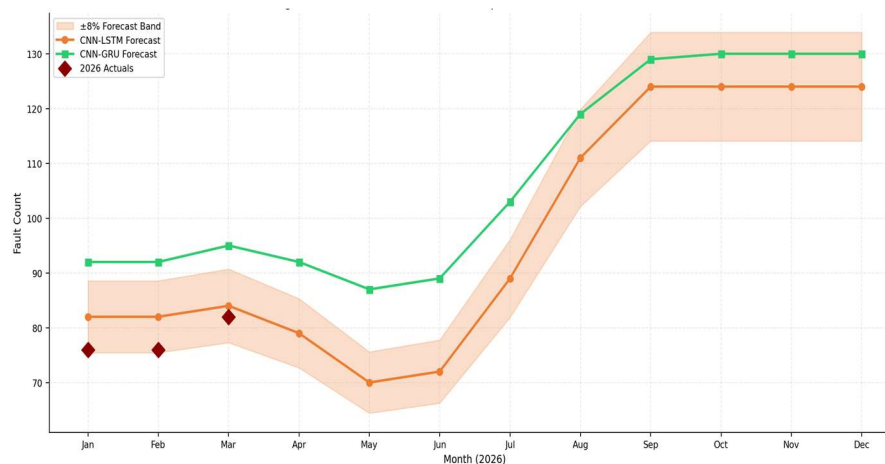


Figure 3. Monthly forecasts for 2026 and analysis of actual results for the first 3 months

Table 2. 2026 Monthly forecast values, risk classification and recommended maintenance mode

| Month          | CNN-LSTM | CNN-GRU | Actual | Risk Level | Maintenance Mode      |
|----------------|----------|---------|--------|------------|-----------------------|
| January 2026   | 82.4     | 91.8    | 76     | HIGH       | Reactive+Preventive   |
| February 2026  | 82.2     | 92      | 76     | HIGH       | Reactive+Preventive   |
| March 2026     | 83.8     | 94.6    | 82     | HIGH       | Preventive            |
| April 2026     | 78.7     | 91.9    | —      | MEDIUM     | Condition Monitoring  |
| May 26         | 70.3     | 86.8    | —      | MEDIUM     | Condition Monitoring  |
| June 2026      | 72       | 88.6    | —      | MEDIUM     | Condition Monitoring  |
| July 2026      | 89.4     | 103     | —      | HIGH       | Intensive Maintenance |
| August 2026    | 111.1    | 118.9   | —      | CRITICAL   | Emergency Maintenance |
| September 2026 | 124.4    | 129.3   | —      | CRITICAL   | Emergency Maintenance |
| October 2026   | 124.5    | 129.8   | —      | CRITICAL   | Emergency Maintenance |
| November 2026  | 124.5    | 129.8   | —      | CRITICAL   | Emergency Maintenance |
| December 2026  | 124.5    | 129.8   | —      | CRITICAL   | Emergency Maintenance |

5.3. Forecast–actual comparison and performance metrics

The deviations between the forecasted and actual values for the January–March 2026 period are illustrated in Figure 4. A notable finding is that the performance of both models was observed to be lower than a simple average baseline in the split sample set. These results are consistent with studies [3] highlighting the limitations of deep learning algorithms when dealing with limited datasets and non-stationary time series.

Table 3. Month-by-month forecast comparison against the simple average baseline — January–March 2026

| Month         | Actual | Simple Avg. Baseline <sup>1</sup> | Error | CNN-LSTM | Error | CNN-GRU | CNN-GRU Error |
|---------------|--------|-----------------------------------|-------|----------|-------|---------|---------------|
| January 2026  | 76     | 60.1                              | −15.9 | 82.4     | +6.4  | 91.8    | +15.8         |
| February 2026 | 76     | 60.1                              | −15.9 | 82.2     | +6.2  | 92.0    | +16.0         |
| March 2026    | 82     | 60.1                              | −21.9 | 83.8     | +1.8  | 94.6    | +12.6         |

Simple Average Baseline: constant prediction equal to the 2018–2024 training period monthly mean (60.1 events/month).  
Error = Predicted – Actual

Table 4. Performance metrics comparison — Validation set (2025) and out-of-sample period (Jan–Mar 2026)

| Metric                       |          | Simple Avg. Baseline | CNN-LSTM | CNN-GRU |
|------------------------------|----------|----------------------|----------|---------|
| Validation Set (2025)        | MAE      | 8.60                 | 14.00    | 15.42   |
|                              | RMSE     | 10.63                | 16.24    | 17.29   |
|                              | MAPE (%) | 15.54                | 26.40    | 27.88   |
| 2026 Out-of-Sample (Jan–Mar) | MAE      | 17.90                | 4.78     | 14.82   |
|                              | RMSE     | 18.12                | 5.22     | 14.90   |
|                              | MAPE (%) | 22.85                | 6.23     | 19.09   |

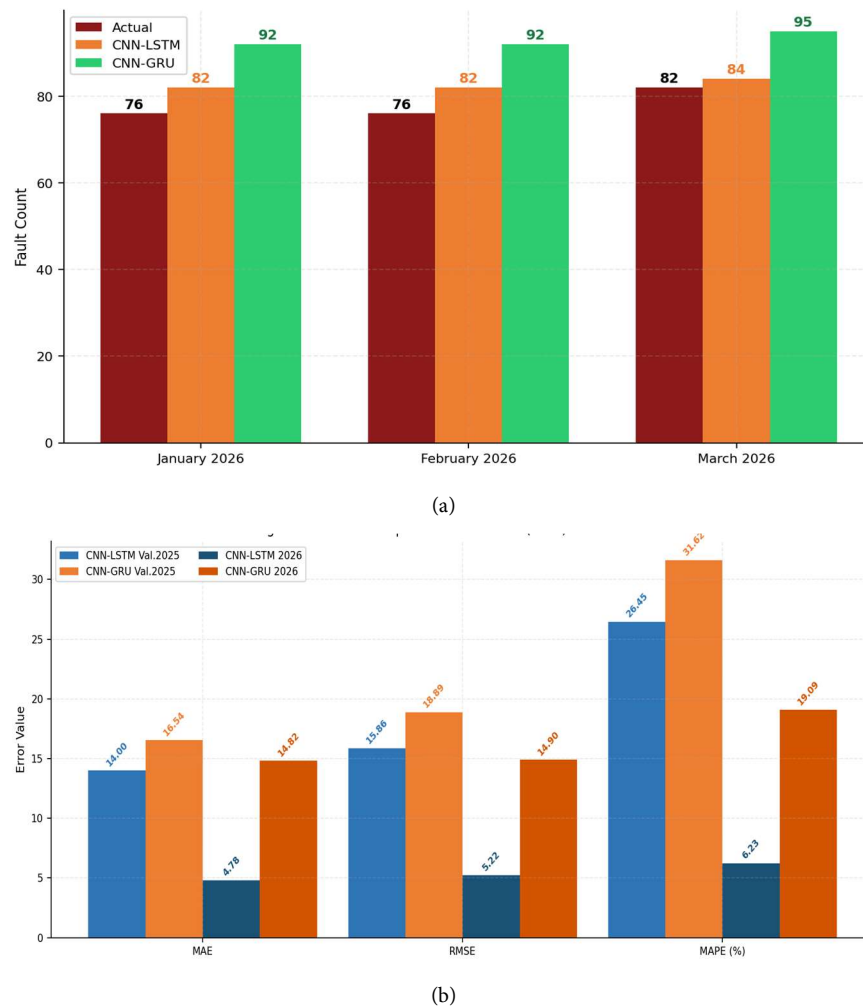


Figure 4. Deviations between forecast and actual values (a) Fault count comparison, (b) Metric comparison.

Bold values indicate the best-performing model for each period and metric.

## 6. Discussion

### 6.1. Interpretation of performance findings

Although the CNN-LSTM model's high over-prediction tendency on the validation set (MAPE = 26.4%) appears at first glance to be a negative indicator, it is noteworthy that this systematic deviation provided an unexpected contribution in terms of model selection: the model reached forecasts (82.4, 82.2, 83.8) that show high agreement with the actual 2026 values (76, 76, 82) and achieved an extremely low 2026 error rate of MAPE = 6.23%. The superiority of CNN-LSTM over CNN-GRU across all metrics indicates that the four-gate architecture internalized the autocorrelation structure of this time series more effectively.

It must be acknowledged that the 2026 out-of-sample evaluation rests on only three months of actual data, which limits the statistical robustness of this comparison; the strong MAPE = 6.23% result should therefore be treated as indicative rather than conclusive. That said, the apparent discrepancy—strong validation error yet accurate 2026 forecasting—is consistent with the phenomenon of distribution shift (or covariate shift) described in deep learning literature [6]. The CNN-LSTM model's systematic over-prediction during 2025 thus reflects a bias toward the expected future mean—precisely the level that 2026 actuals converged to. This behaviour, while undesirable in classical validation, constitutes an implicit drift adaptation that benefited out-of-sample performance. Classical time-series baselines such as ARIMA or simple moving averages are structurally constrained to extrapolate from the most recent observations or a fixed historical

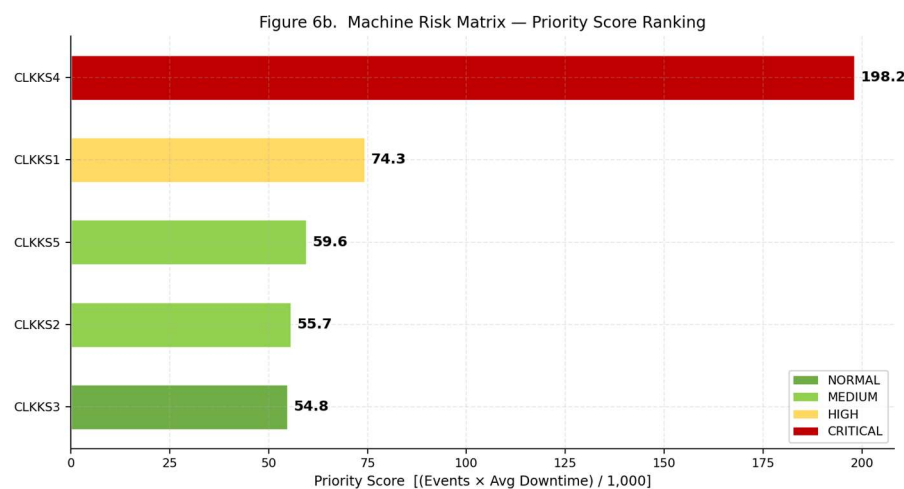
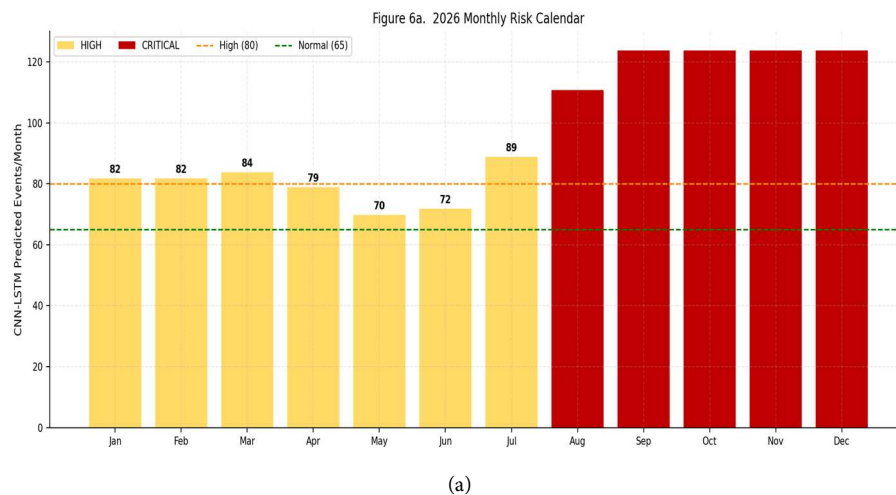
mean, and therefore cannot encode the directional distributional shift that separates the 2018–2024 training regime from the elevated fault levels observed in 2026 [3, 19]. By contrast, the CNN-LSTM’s convolutional feature-extraction stage captures non-linear regime patterns across the full eight-year training window, enabling the model to implicitly weight long-range structural trends over short-term fluctuations and thereby anticipate the upward distributional shift that ARIMA-class models, anchored to local stationarity assumptions, would systematically miss.

## 6.2. Limitations and future research directions

The primary limitations of this study are: the univariate input structure, the 96-month sample depth, and the actual-event window covering only January–March 2026 (with March representing a full calendar month). Future studies could qualitatively improve model performance through the integration of vibration and temperature data obtained from Internet of Things (IoT) sensors, the inclusion of Transformer architectures in the comparison, and the investigation of Bayesian approaches that produce probabilistic confidence bands.

## 7. Predictive Maintenance Action Plan (April–December 2026)

In this section, an operational maintenance implementation schedule was constructed by blending CNN-LSTM forecast outputs with the long-term event patterns of the 2018–2025 period. The plan consists of four sub-components: a risk-thresholding framework, machine-level priority classification, a phased activity programme, and trackable performance targets, following the structured approach outlined by Mobley [20].



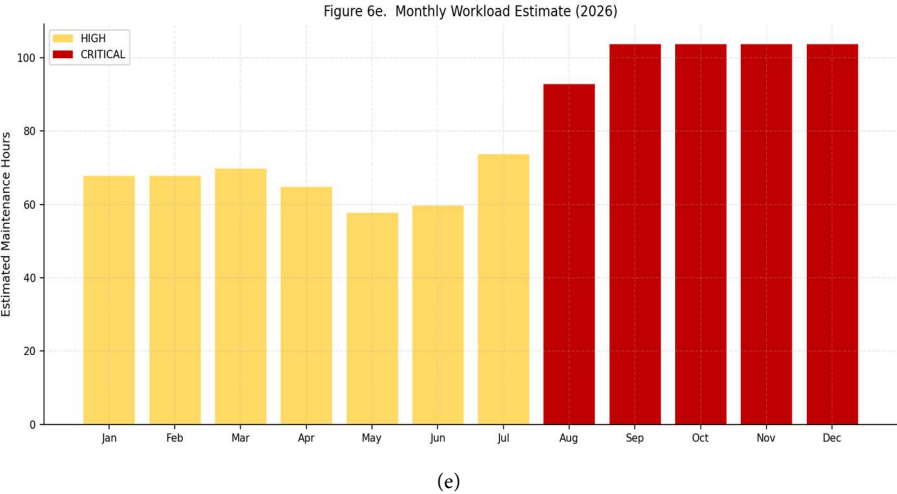
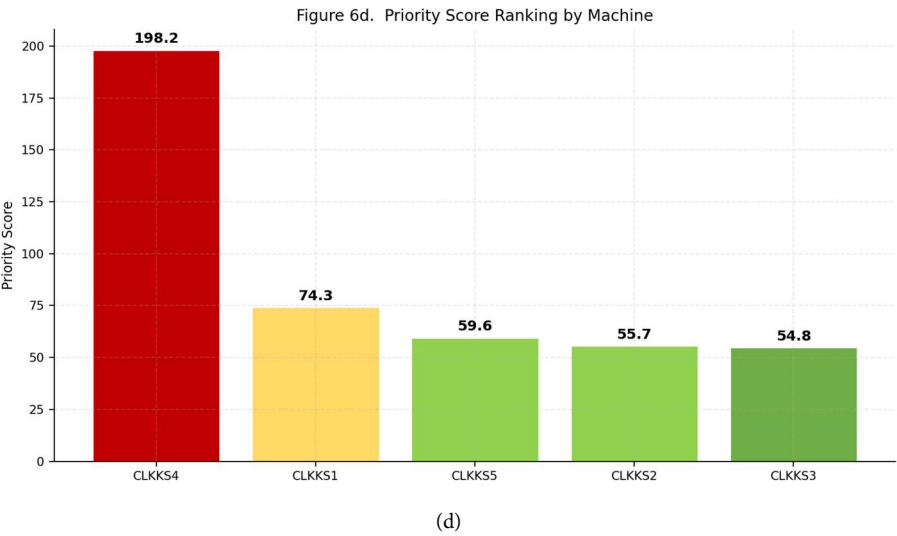
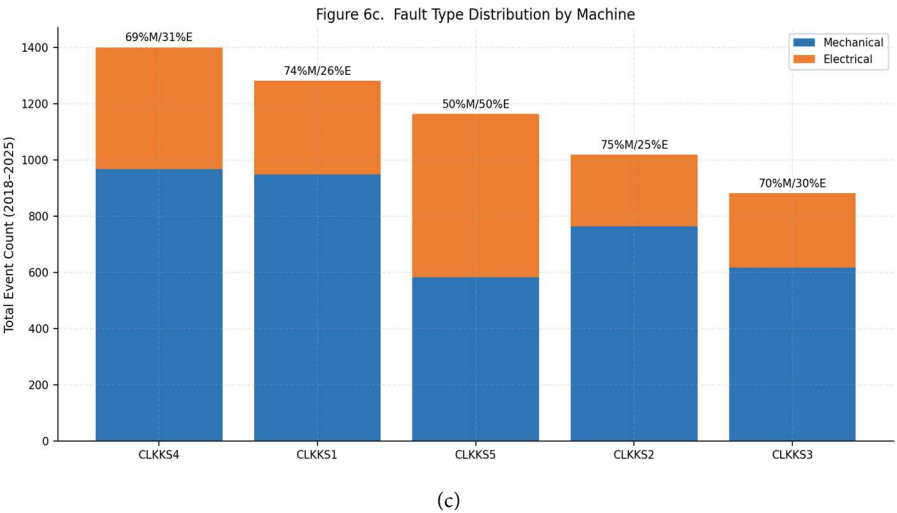


Figure 6. Operational maintenance implementation program, (a) 2026 Monthly risk calendar (CNN-LSTM Forecast), (b) Machine risk matrix-priority score ranking, (c) Fault type distribution by machine (2018–2025), (d) Priority score ranking by machine, (e) Monthly workload estimate (2026).

### 7.1. Risk thresholding framework

CNN-LSTM monthly forecasts were divided into four risk bands:

- Critical risk ( $\geq 105$  events/month): Emergency maintenance readiness, immediate intervention planning, and maximum resource allocation.
- High risk ( $\geq 80$  events/month): Intensive preventive inspection and standby spare capacity,
- Medium risk (65–79 events/month): Condition monitoring and component life management,
- Normal risk ( $< 65$  events/month): Standard schedule and model improvement activities.

It should be noted that the CRITICAL classifications for October, November, and December 2026 are based on the autoregressive plateau value of 124.5 events/month, which reflects the model's long-run mean convergence rather than independently derived monthly estimates (see Section 4.3). Despite this methodological caveat, the plateau value legitimately exceeds the  $\geq 105$  threshold on its own merit, confirming the CRITICAL designation on substantive grounds. Maintenance teams are nonetheless advised to reassess the risk band assignment dynamically as each month's actual data become available through the CMMS monitoring protocol (Table 5), and to consider a downgrade to HIGH status if three consecutive months record fewer than 105 events.

### 7.2. Machine priority classification

The priority score for each machine is determined by Eq. (10);

$$\text{Priority Score} = (\text{Total Event Count} \times \text{Average Downtime}) / 1000 \quad (10)$$

Table 5 presents the ranking of all machines and recommended maintenance approaches.

Table 5. Machine priority matrix and maintenance strategy recommendations

| Machine | Events | Avg. DT   | Priority Score | Risk Class | Dom. Fault  | Approach             |
|---------|--------|-----------|----------------|------------|-------------|----------------------|
| CLKKS4  | 1,401  | 141.5 min | 198.2          | CRITICAL   | Mech. (69%) | Intensive Predictive |
| CLKKS1  | 1,282  | 58.0 min  | 74.3           | HIGH       | Mech. (74%) | Planned Preventive   |
| CLKKS5  | 1,164  | 51.2 min  | 59.6           | MEDIUM     | Mixed       | Periodic Inspection  |
| CLKKS2  | 1,019  | 54.6 min  | 55.7           | MEDIUM     | Mech. (75%) | Periodic Inspection  |
| CLKKS3  | 882    | 62.1 min  | 54.8           | NORMAL     | Mech. (70%) | Standard Schedule    |

CLKKS4, with a score of 198.2, is the most critical piece of equipment in the facility; this machine accounts for approximately 45% of total downtime, and 69% of its faults are mechanical in origin.

### 7.3. Three-phase implementation

*Short Term: High-Risk Period (April–June 2026)*

- CLKKS4 comprehensive overhaul: Bearings and housings, power-transmission gears, sealing elements; dimensional inspection of all wear surfaces
- Weekly vibration and thermal scanning: Off-line and on-line measurements across the facility for early detection of anomaly trends
- Spare parts buffer: Critical spare parts for CLKKS4 and CLKKS1 must be secured; items with long lead times should be ordered in advance

*Medium Term: Transition and Critical Period (July–October 2026)*

- CLKKS1 and CLKKS5 component renewal: Motor windings, sensor calibrations, and systematic inspection of all cables

and connection points

- CMMS integration: Automatic triggering of work orders, enrichment of historical records, and improvement of model input quality

*Long Term: Improvement Period (November–December 2026)*

- CNN-LSTM retraining: Adding 2026 Q1–Q3 data to the training set to increase sample size and compensate for model distribution shift
- IoT sensor pilot installation: Installing at least a vibration accelerometer and an infrared thermal sensor on CLKKS4
- Multivariate feature expansion: Fault category, downtime, and seasonal sinusoidal indicators will be introduced to the model as additional inputs

| Table 6. Predictive maintenance three-phase implementation Schedule |              |          |                                                                                                                      |                |                     |
|---------------------------------------------------------------------|--------------|----------|----------------------------------------------------------------------------------------------------------------------|----------------|---------------------|
| Phase                                                               | Period       | Risk     | Core Activities                                                                                                      | Unit           | Success Criterion   |
| SHORT TERM                                                          | Apr–Jun 2026 | HIGH     | CLKKS4 complete overhaul; weekly vibration & thermal scanning; critical spare buffer                                 | Maint. Mgr.    | Downtime ≤90 min    |
| MEDIUM TERM                                                         | Jul–Oct 2026 | MED–CRIT | CLKKS1 & CLKKS5 component renewal; Computerised Maintenance Management System (CMMS) integration; model update cycle | Maint.+Quality | Monthly events < 65 |
| LONG TERM                                                           | Nov–Dec 2026 | CRIT→Std | CNN-LSTM retraining with 2026 data; IoT sensor pilot; 2027 budget optimisation                                       | Eng. + IT      | MAPE < 35%          |

7.4. Performance monitoring indicators (KPIs)

The performance evaluation framework for the 2026 period, as detailed in the provided dataset, reflects a strategic shift toward enhanced operational reliability and cost-efficiency. By targeting a reduction in monthly fault events from approximately 78 to fewer than 65, alongside a significant improvement in average response times from 77 to below 60 minutes, the proposed plan prioritizes system agility. This optimization is further supported by the objective to reduce the CLKKS4 event share to below 18% and the establishment of a 60/40 planned-to-unplanned maintenance ratio, which was previously unmeasured. Despite a high initial forecast accuracy (MAPE) of 6.23% in Q1 2026, the long-term monitoring strategy adopts a broader threshold of <35% for the final quarter to accommodate potential model variances, ultimately aiming for a 15% reduction in total maintenance costs compared to the base period.

Table 7. Comparative analysis of operational KPIs: 2025 baseline versus 2026 strategic targets and monitoring methodologies.

| KPI                        | Reference (2025)     | 2026 Target      | Monitoring Method       |
|----------------------------|----------------------|------------------|-------------------------|
| Monthly Total Fault Events | ≈78 events/month     | <65 events/month | Fault management system |
| Average Response Time      | 77 min/event         | <60 min/event    | CMMS reports            |
| CLKKS4 Event Share         | 24% (facility total) | <18%             | Machine-level indicator |
| Planned/Unplanned Ratio    | Not measured         | >60/40           | Work order tracking     |
| Forecast Accuracy (MAPE)   | 6.23% (2026 Q1)      | <35% (2026 Q4)   | Quarterly model eval.   |
| Total Maintenance Cost     | Base Period          | –15%             | Cost accounting system  |

8. Conclusion

This study presents an integrated methodology that combines eight years of CLKKSM facility fault records with a data-driven predictive maintenance framework. In the experimental comparison, the CNN-LSTM model consistently outperformed the CNN-GRU by forecasting the actuals of the first quarter of 2026 with MAE = 4.78 and MAPE = 6.23%.

The three-phase action schedule designed in light of the forecasts obtained is built on the following main axes: intensive predictive maintenance allocation to CLKKS4 (priority score 198.2); comprehensive overhauls and early warning programmes for the April–June high-risk period; CMMS integration and component renewal for the July–October critical period; and model retraining and an IoT sensor pilot for the November–December improvement period.

Future work can advance along two main lines: (i) multivariate architectural expansion covering machine operation parameters, environmental conditions, and maintenance history; and (ii) the integration of Bayesian methods providing probabilistic confidence bands, or Transformer-based architectures that constitute strong references in sequence data learning.

## Conflict of Interest Statement

The authors declare that there is no conflict of interest

## Authorship Contribution Statement

**S.E:** Conceptualization, Methodology, Software, Supervision; **O.T:** Validation, Investigation, Writing – Original Draft; **Ç.B:** Formal analysis, Visualization, Project administration, Writing – Review & Editing.

## References

- [1] A. K. Jardine, D. Lin, and D. Banjevic, "A review on machinery diagnostics and prognostics implementing condition-based maintenance," *Mechanical systems and signal processing*, vol. 20, no. 7, pp. 1483-1510, 2006. doi: 10.3224/güzelbişi.21331.454
- [2] J. Lee, F. Wu, W. Zhao, M. Ghaffari, L. Liao, and D. Siegel, "Prognostics and health management design for rotary machinery systems—Reviews, methodology and applications," *Mechanical systems and signal processing*, vol. 42, no. 1-2, pp. 314-334, 2014. doi: 10.1016/j.ymssp.2013.06.004
- [3] W. Zhang, D. Yang, and H. Wang, "Data-driven methods for predictive maintenance of industrial equipment: A survey," *IEEE systems journal*, vol. 13, no. 3, pp. 2213-2227, 2019. doi: 10.1109/JSYST.2019.2905565
- [4] S. Hochreiter, *Long short-term memory*, Neural Computation MIT-Press, 1997.
- [5] K. Cho et al., "Learning Phrase Representations Using RNN Encoder–Decoder for Statistical Machine Translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Oct. 25–29, 2014, Doha, Qatar, pp. 1724–1734. Stroudsburg, PA, USA: Association for Computational Linguistics (ACL), 2014.
- [6] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436-444, 2015. doi:10.1038/nature14539
- [7] R. Zhao, R. Yan, J. Wang, and K. Mao, "Learning to monitor machine health with convolutional bi-directional LSTM networks," *Sensors*, vol. 17, no. 2, p. 273, 2017. doi:10.3390/s17020273
- [8] Y. Wu, M. Yuan, S. Dong, L. Lin, and Y. Liu, "Remaining useful life estimation of engineered systems using vanilla LSTM neural networks," *Neurocomputing*, vol. 275, pp. 167-179, 2018. doi: 10.1016/j.neucom.2017.05.063
- [9] X. Li, Q. Ding, and J.-Q. Sun, "Remaining useful life estimation in prognostics using deep convolution neural networks," *Reliability Engineering & System Safety*, vol. 172, pp. 1-11, 2018. doi: 10.1016/j.ress.2017.11.021
- [10] A. Vaswani et al., "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.
- [11] A. Zeng, M. Chen, L. Zhang, and Q. Xu, "Are Transformers Effective for Time Series Forecasting?," in *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence (AAAI-23)*, Feb. 7–14, 2023, Washington, DC, USA, vol. 37, no. 9, pp. 11121–11128. Washington, DC, USA: Association for the Advancement of Artificial Intelligence (AAAI), 2023. doi: 10.1609/aaai.v37i9.26317
- [12] Y. Lei, B. Yang, X. Jiang, F. Jia, N. Li, and A. K. Nandi, "Applications of machine learning to machine fault diagnosis: A review and roadmap," *Mechanical systems and signal processing*, vol. 138, p. 106587, 2020. doi: 10.1016/j.ymssp.2019.106587

- [13] R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang, and R. X. Gao, "Deep learning and its applications to machine health monitoring," *Mechanical systems and signal processing*, vol. 115, pp. 213-237, 2019. doi: 10.1016/j.ymssp.2018.05.050
- [14] R. Zhao, D. Wang, R. Yan, K. Mao, F. Shen, and J. Wang, "Machine health monitoring using local feature-based gated recurrent unit networks," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 2, pp. 1539-1548, 2017. doi: 10.1109/TIE.2017.2733438
- [15] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional Block Attention Module," in *Proceedings of the 15th European Conference on Computer Vision (ECCV 2018)*, Sept. 8-14, 2018, Munich, Germany, pp. 3-19. Cham, Switzerland: Springer, 2018. doi: 10.1007/978-3-030-01234-2\_1
- [16] H. Yu, C. Chen, N. Lu, and C. Wang, "Deep auto-encoder and deep forest-assisted failure prognosis for dynamic predictive maintenance scheduling," *Sensors*, vol. 21, no. 24, p. 8373, 2021. doi: 10.3390/s21248373
- [17] Q. Wen et al., "Time series data augmentation for deep learning: A survey," *arXiv preprint arXiv:2002.12478*, 2020. doi: 10.24963/ijcai.2021/631
- [18] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014. doi: 10.48550/arXiv.1412.6980
- [19] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.
- [20] R. K. Mobley, *An introduction to predictive maintenance*. Elsevier, 2002.