International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V73I8P116

Volume 73 Issue 8, 190-201, August 2025
© 2025 Seventh Sense Research Group®

Original Article

Machine Learning Applications for Condition-Based Maintenance of Critical Machinery Components

Rindi Kusumawardani[1], Nani Kurniati[2], Fauzi Irfandi Yusuf[3]

[1,2,3]*Industrial and Systems Engineering Department, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia.*

[1]*Corresponding Author : rindi@its.ac.id*

Received: 24 January 2025 Revised: 06 August 2025 Accepted: 12 August 2025 Published: 30 August 2025

Abstract - *This study evaluates the performance of Machine Learning (ML) and Deep Learning (DL) models for Condition-Based Maintenance (CBM) of a raw mill, focusing on motor condition prediction. The work systematically analyzes and compares the predictive capabilities of Random Forest (RF) and Long Short-Term Memory (LSTM) models for three critical motors, Main Drive (MD), Separator (SR), and Fan (FN), using varying input timespans to identify the optimal approach. The RF model consistently outperformed the LSTM model across all motors, achieving superior accuracy and efficiency. Specifically, the RF model recorded an RMSE of 0.0235 and an R² of 97.9% for the MD motor, compared to the LSTM model's RMSE of 0.0858 and R² of 72.2%. Similarly, the RF model achieved an RMSE of 0.0466 and an R² of 91.1% for the SR motor, outperforming the LSTM model. While the LSTM model offers higher configurability and is well-suited for complex time-series tasks, its performance in this study was hindered by limited and noisy data, underscoring the robustness of shallow ML models like RF in such scenarios. A MATLAB-based dashboard was developed to visualize motor conditions, predict accuracy, and allow real-time updates. These results highlight RF as the preferred approach for CBM in industrial settings, offering greater consistency, computational efficiency, and ease of implementation compared to its deep learning counterpart.*

Keywords - *Raw mill, Condition Based Maintenance (CBM), Prognostics, Random Forest (RF), Long Short-Term Memory (LSTM), Machine Learning, Deep Learning.*

1. Introduction

The cement industry is a critical sector with significant contributions to infrastructure development [1]. Cement production typically involves multiple strategically located plants equipped with advanced facilities to ensure efficient operations. The production process begins with raw material acquisition and premixing. Materials such as clay, limestone, silica sand, and iron sand are extracted from quarries or transported via jetties and stored in stockpiles. These materials are then processed in raw mills, which are ground and mixed into raw meals. The raw meal is subjected to high temperatures in preheaters and kilns, transforming it into clinker. Finally, the clinker is ground into fine cement powder in cement mills. This process ensures the material meets the required quality and granular size before being shipped for use. Despite advancements in production systems, equipment downtime remains a significant challenge for cement plants. Downtime in critical components, such as the raw mill system, can considerably affect overall plant efficiency and production capacity [2]. For instance, an analysis of a raw mill system in one cement plant from 2021 to early 2023 revealed availability ranging from 83.95% to as low as 57.21%, with an average of 72.46%, falling short of the ideal 90% standard [3]. This inefficiency leads to reduced productivity, longer lead times, and diminished customer satisfaction [4]. Raw mill systems are comprised of multiple components, including the mill, separators, fans, and transport systems. As a critical element, motors are essential power functions like the raw mill table and fans, making them pivotal for continuous operations. Tool condition is a significant parameter that directly influences the performance and quality of the machining process, underscoring the importance of continuous monitoring for efficient operation and quality control in industrial systems [5]. Conventional maintenance strategies, such as scheduled overhauls and manual interpretation of sensor data for temperature and vibration monitoring, often fail to prevent unexpected failures or optimize maintenance schedules. These limitations highlight the need for advanced predictive approaches like Condition-Based Maintenance (CBM). CBM utilizes real-time equipment condition data for diagnostics and prognostics, predicting failures before they occur [4, 6]. CBM refers to identifying anomalies and diagnosing faults. Anomaly detection focuses on determining whether the system is operating within normal parameters, while fault diagnosis aims to identify deviations from expected behavior and evaluate their severity [7]. In general, the primary goal of CBM is to do a real-time assessment of equipment conditions in order to make maintenance decisions, therefore reducing unnecessary maintenance and associated costs [8]. However, traditional CBM approaches often rely on rule-based or preventive maintenance strategies, which can lead to inefficiencies such as under-maintenance or over-maintenance. These conventional methods struggle to adapt to dynamic industrial conditions and are less effective in handling the complexity of high-dimensional sensor data.

Several studies have explored the use of various Machine Learning (ML) algorithms for predictive maintenance. Deep Learning (DL), a subfield of ML, has also demonstrated its effectiveness in such tasks. Research on production line maintenance has shown that Random Forest (RF) algorithms provide highly accurate predictions for prognostic assessments [9]. Random Forest models have been found to perform well regardless of dataset size or binning methods, even in applications involving ball bearings [10]. Additionally, Long-Short Term Memory (LSTM) networks, a type of Deep Learning algorithm, have been favored for predictive maintenance due to their unique design. LSTM networks use neural networks as memory cells, which are well-suited for time-series data. Studies involving LSTM in predictive maintenance for assets like those of the Spanish Naval Army have yielded promising results [11]. Further research has demonstrated the application of LSTM models in predicting material crack growth in aluminum, achieving over 95% accuracy [12].

Despite advances in both shallow machine learning and deep learning for predictive maintenance, existing studies rarely evaluate and compare RF and LSTM models on the same noisy, high-dimensional industrial sensor datasets. RF is favored in industrial fault diagnosis for its robustness to noise and low computational overhead [13]. By contrast, LSTM excels at modeling temporal dependencies but demands large, clean datasets and substantial computational resources, which can impede real-time implementation [14]. Furthermore, most prognostic approaches focus solely on one-step-ahead forecasts, neglecting the practical need for multi-horizon predictions (e.g., 1–60 minutes) that maintenance teams use for planning. Single-step predictions in prognostics often neglect uncertainty, leading to unreliable forecasts [15-17]. Consequently, there remains a gap in head-to-head, multi-horizon comparisons of RF and LSTM on real-world industrial data rather than benchmark or simulated datasets.

This research proposes the application of CBM to enhance the maintenance strategy for raw mill systems by leveraging AI-based models. A comparative study between RF and LSTM algorithms will identify the best approach for predicting equipment health under varying data conditions. For high-dimensional datasets, DL typically outperforms shallow ML, while for low-dimensional datasets, shallow ML offers better interpretability and performance [18]. Sensor data, collected over 26 months, is pre-processed through outlier removal, missing value handling, MinMax scaling, and Principal Component Analysis (PCA) [19, 20]. Data is split into training and testing sets, and both RF and LSTM models are trained and optimized through hyperparameter tuning. Model performance is evaluated using metrics like Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and R^2, with the best model selected for use [21]. Additionally, a dashboard interface will be developed that follows seven principles [22] to visualize predictions and sensor data, providing actionable insights for maintenance scheduling, spare part procurement, and production planning. The proposed approach aims to enhance operational reliability, reduce maintenance costs, and optimize production efficiency, contributing to the industry's overall growth and competitiveness.

2. Data Acquisition and Pre-Processing

2.1. Data Acquisition

The raw mill system is equipped with two bearing vibration sensors, two bearing temperature sensors, and three winding temperature sensors on the Main Drive motor (MD), Separator Motor (SR), and Fan Motor (FN). The main drive motor operated stably below the trip threshold, while winding temperatures showed minor variations. The non-drive end bearing experienced higher temperatures than the drive end bearing, with both bearings surpassing the alarm threshold for vibrations. The separator motor also remained within acceptable thresholds, and its bearings mostly maintained acceptable vibration levels. Similarly, the fan motor operated well within alarm and trip thresholds, with slight variations in winding temperatures and vibration levels between drive-end and non-drive-end bearings.

The dataset comprises 26 months of one-minute-interval sensor readings collected from a raw mill system at a cement production plant. Three motors, the MD, SR, and FN, were each instrumented with seven sensors (two bearing vibration, two bearing temperature, and three winding temperature), yielding over 1.133.286 data points per sensor and approximately 7.9 million rows in total. As raw data sources often contain noise and inconsistencies due to automatic data collection flaws, data cleaning is applied to improve data quality and provide accurate and reliable data for further analysis [23].

2.2. Data Cleaning

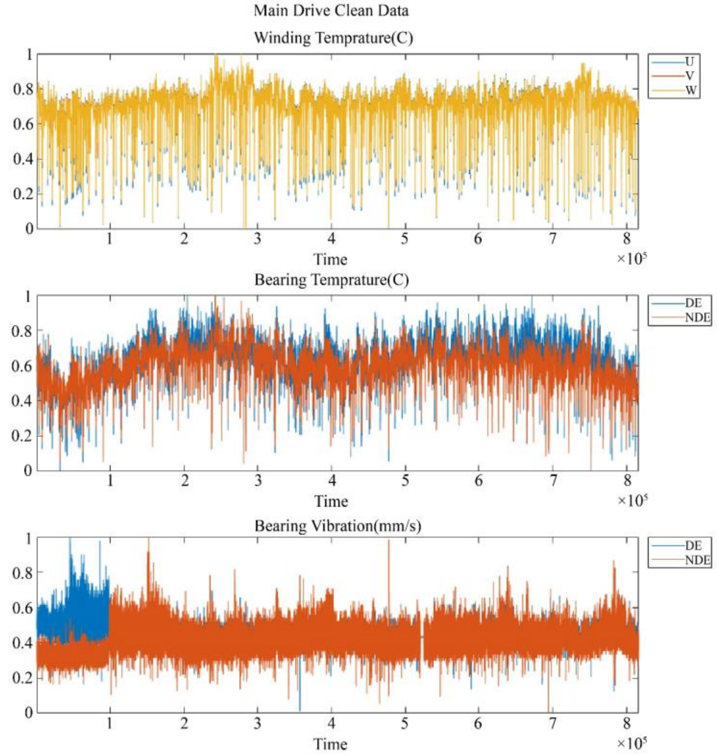
The data cleaning process begins by removing sensor readings recorded while the system was powered off to ensure accuracy in subsequent condition predictions. Although the dataset does not have missing values, outliers are detected and eliminated using the Hampel filter with a moving median method and a window of 60. Addressing outliers is critical for maintaining data integrity, as noisy and irregular data points can distort analysis results. In cases where gaps in the data arise from sensor failures, interpolation techniques like cubic spline interpolation are applied. This ensures data continuity, transforming incomplete datasets into reliable, comprehensive inputs for further analysis. By addressing these issues, the cleaned dataset becomes suitable for predictive maintenance modeling, offering a robust foundation for accurate and actionable insights [24]. The cleaned data is presented in Figure 1. After cleaning and scaling, the dataset is reduced to 816,783 rows, representing 72.07% of the original raw data. Despite some sensors showing prolonged constant values, these readings are assumed to be accurate and are retained for analysis. This rigorous cleaning process ensures high-quality data for reliable predictive modeling.

2.3. Feature Extraction

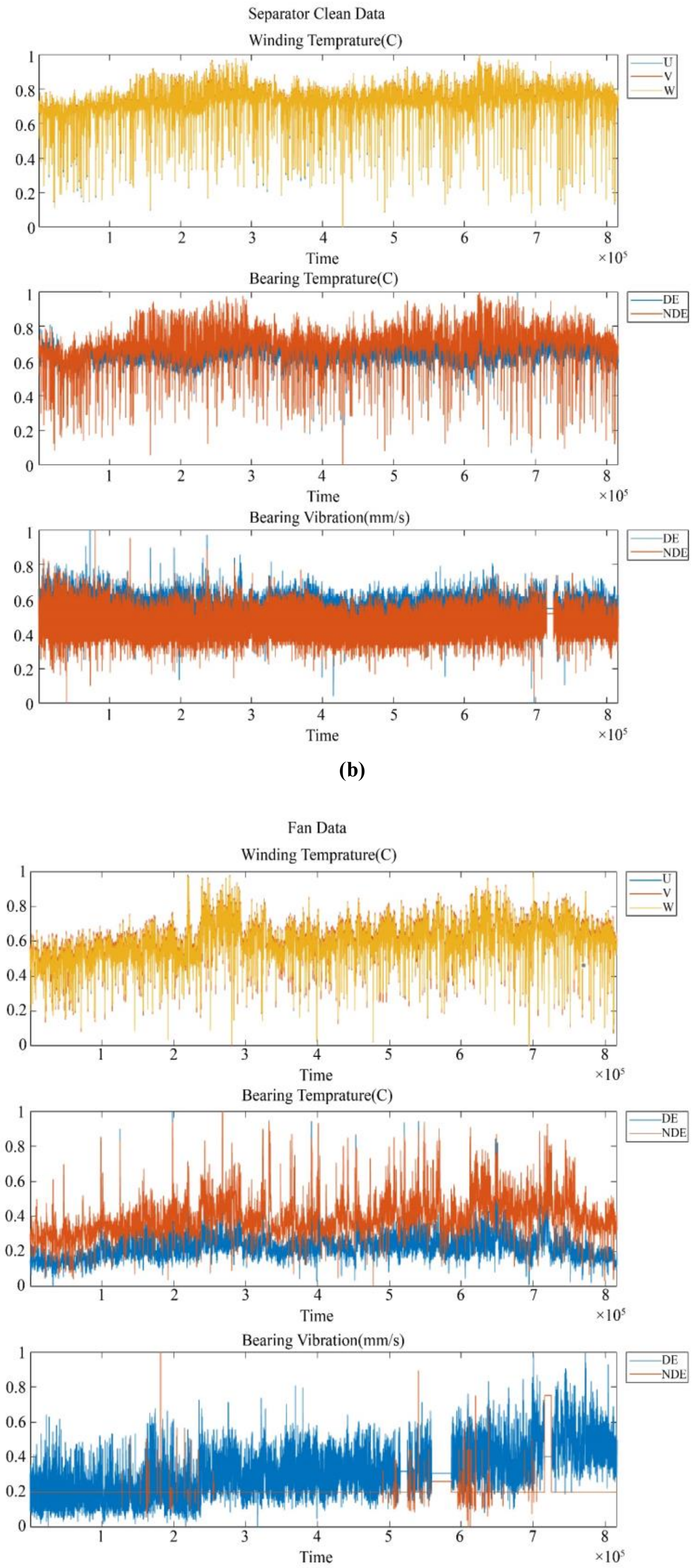
After cleaning the dataset, Principal Component Analysis (PCA) is applied to extract key features, summarizing the data efficiently without significant information loss. PCA is particularly effective in time-series forecasting and automatically selects the principal component with the highest explained variance for further analysis [20].

The data is prepared for prediction, where the current time step (x_t) serves as the output, and previous time steps (x_{t-1}, x_{t-2}, x_{t-3}, x_{t-4},..., x_{t-n}) are used as inputs. Various historical time spans are tested to identify the optimal input size that balances prediction accuracy and computational efficiency. This approach ensures the model is well-trained to forecast future conditions based on relevant historical patterns. The cleaned dataset is divided into training (80%) and testing (20%) subsets, with 653,427 rows for training and 163,356 rows for testing. PCA is applied separately to the training and testing datasets to mimic real-world scenarios, where unseen test data represents future sensor readings. The number of principal components to retain is determined using established criteria. The Kaiser criterion retains components with eigenvalues greater than 1 [25], while the threshold method uses a cutoff of 70% for total variance explained [26].

The PCA biplot shown in Figure 2 illustrates a dense spread of data among condition sensor readings, likely due to high inter-variable correlation. The first principal component effectively captures most of the data variance compared to the second component. The data projected onto the selected principal component summarizes motor conditions, which are then used for predicting future conditions of the raw mill system motors. The condition data is configured such that previous conditions serve as input, and the current condition serves as the output for machine learning and deep learning algorithms. In time series data, the number of previous observations is vital in capturing the underlying relationship in the time series. Too many input nodes tend to overfit the training data, while too few input nodes might lead to underfitting the data [27]. The input that is to be tested is at n equal to 1, 5, 10, 20, 40, 60. The input selection is multiplied over to cover the possibility of overfitting or underfitting the data while preserving the computational efficiency required. The input is capped to 60, as a higher input indicates an hourly interval that would be easier to capture using an hourly interval instead of a minute interval.



(a)



(b)

(c)

Fig. 1 Clean data, (a) Main drive motor, (b) Separator motor, and (c) Fan motor.

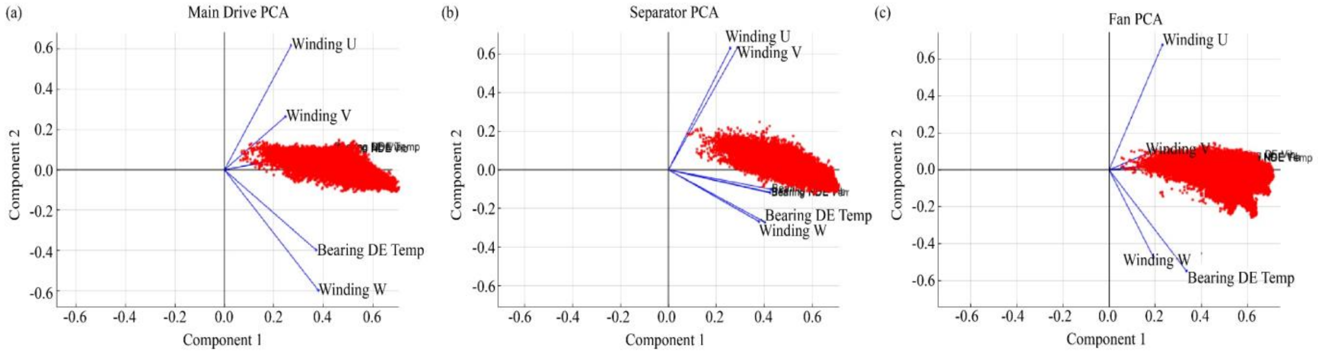


Fig. 2 PCA Biplot for (a) Main Drive Motor (MD), (b) Separator Motor (SR), and (c) Fan Motor (FN).

3. Model Building

The initial hyperparameter settings for both the Random Forest (RF) and Long Short-Term Memory (LSTM) algorithms are derived from prior research and MATLAB's default configurations. For RF, the primary hyperparameters include the number of learning cycles, the number of predictors at each split, leaf size, and maximum number of splits. For LSTM, the key hyperparameters are the LSTM hidden units, epochs, number of LSTM layers, and dropout value.

3.1. Random Forest Model Building

The initial model employs an RF machine learning algorithm, trained and tested using pre-split data. The default hyperparameter settings provided in MATLAB are used during the initial training phase [28]. The RF model's prediction accuracy is evaluated across different input timespans using metrics such as RMSE, MAPE, Mean Absolute Error (MAE), and R^2. For the main drive motor, the optimal input timespans are 10 and 40, with an input of 10 yielding an RMSE of 0.0235 and an R^2 of 97.9%, while an input of 40 achieves the lowest MAE (0.0111) and MAPE (0.7285).

For the separator motor, the best timespans are 10 and 60, where an input of 10 provides an RMSE of 0.0237 and an R^2 of 97.6%, and an input of 60 results in the lowest MAE (0.0113) and MAPE (0.6795). Similarly, for the fan motor, timespans of 10 and 40 are most effective, with an input of 10 achieving an RMSE of 0.0221 and an R^2 of 98.1%, while an input of 40 produces the best MAE (0.0127) and MAPE (0.9943).

Despite each motor having two optimal input timespan options, the timespan of 10 is selected for all three motors due to its computational efficiency. Inputs yielding the best R^2 demonstrate the model's ability to predict a higher proportion of variance in the data, while those with the lowest RMSE reflect minimal prediction error on the same scale as the data [29]. However, no definitive agreement exists on whether RMSE or MAE is the most appropriate metric for error representation [30].

Based on the selected input timespans, the RF model undergoes optimization using MATLAB's Bayesian auto optimizer over 10 iterations. Optimization enhances the performance of the RF models applied to the separator motor and fan motor, while the main drive motor's unoptimized model outperforms the optimized version in terms of RMSE, MAE, and MAPE.

This discrepancy is attributed to overfitting, where the model overemphasizes noise in the training data rather than capturing its underlying patterns [31]. Overall, the Bayesian optimization process effectively improves the RF models for the separator and fan motors, yielding enhanced predictive accuracy.

3.2. LSTM Model Building

The LSTM algorithm was selected as the deep learning model for time series prediction. Its default hyperparameters included 128 hidden LSTM units, 30 epochs, 64 hidden units in the dense layer, and a dropout rate of 0.1.

The model architecture comprised five layers as shown in Figure 3, including an input layer, LSTM hidden layers incorporating long-term memory weights, a dropout layer to prevent overfitting, two dense layers for data processing, and a ReLU activation function for final output predictions. Performance evaluation of the LSTM model was conducted on three motor types: the main drive motor, separator motor, and fan motor, using different input time spans.

For the main drive motor, an input timespan of five yielded the best results with an RMSE of 0.0858, MAE of 0.0678, MAPE of 4.1197, and R^2 of 72.2%. In contrast, for the separator and fan motors, the optimal input timespan was one, achieving RMSE values of 0.0466 and 0.0436, MAE values of 0.0284 and 0.0309, MAPE values of 1.6722 and 2.2505, and R^2 values of 91.1% and 92.7%, respectively.

Generally, smaller input timespans resulted in better performance, leading to the selection of an input timespan of one for the separator and fan motors and an input timespan of five for the main drive motor.

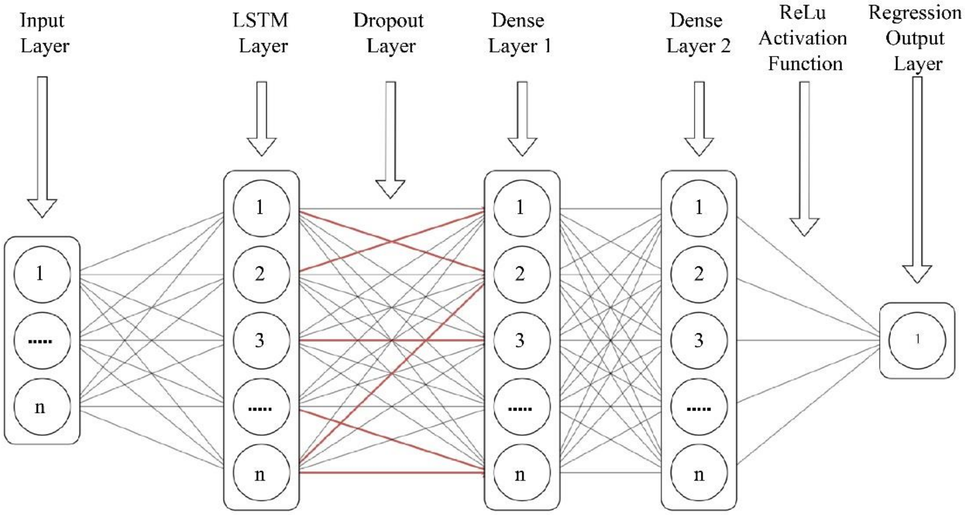


Fig. 3 Deep learning layer structure

The LSTM models were optimised using MATLAB's experiment manager with a Bayesian optimizer, performing 10 iterations for each model. The validation data comprised 10% of the training dataset. While the optimized model for the main drive motor demonstrated improved performance following hyperparameter tuning, the optimized models for the separator and fan motors exhibited a decline in performance compared to their unoptimized counterparts.

This decline suggests potential overfitting, likely due to the Bayesian optimizer's emphasis on validation data performance, which resulted in models excelling on validation data but failing to generalize effectively to test data.

4. Result and Discussion

4.1. Model Evaluation and Comparison

In comparing ML models with DL models, the RF model outperformed the LSTM model in predicting motor conditions. Table 1 provides a general performance summary between the RF and LSTM models. The RF model consistently used the same input timespan to achieve the best performance across all motors, whereas the LSTM model required different input timespans for each motor. This discrepancy in input timespans can be attributed to the greater configurability of the DL model, which enables more extensive customization of hyperparameters and neural network architecture.

Table 1. General performance summary between the RF and the LSTM model

Motor	Model	Input Timespan	RMSE	MAE	MAPE	R^2
MD	RF	10	0,0235	0,0113	0,7356	0,9790
	LSTM	5	0,0479	0,0240	1,6247	0,9143
SR	RF	10	0,0237	0,0114	0,6836	0,9765
	LSTM	1	0,0466	0,0284	1,6722	0,9109
FN	RF	10	0,0221	0,0128	1,0010	0,9811
	LSTM	1	0,0436	0,0309	2,2505	0,9278

Based on the general performance of RF and LSTM models applied to the motors, the RF model achieved better performance in the prediction of motor condition. The RF model applied to the motors consistently has the same input timespan for the best prediction performance for 10 previous input timespans. For LSTM models, less input timespan is required to get the best output. For the main drive motor, the best input timespan is five minutes, while for separator and fan motors, the best input is one minute after the previous condition. The difference in input timespan consistency is based on the difference in configurability between the RF and LSTM models. The RF is configurable to the extent of configuring the hyperparameters of the algorithm, which can be optimized using the Bayesian hyperparameters. The limited capability of the RF model to be configured results in the consistency of performance and results between the main drive motor, separator motor, and fan motor. Deep learning models, on the other hand, are highly configurable. Deep learning models can be configured with more than the available hyperparameters. The layers of neural networks, the types of neural networks in each layer, the activation functions, and other complementary functions, including dropout layers. Thus, the best achieved input timespan for the LSTM model differs for the main drive motor, separator motor, and fan motor. The configurability of the deep learning model also contributes to the general performance compared to the RF model. For the acquired dataset and specific LSTM structure applied, the RF model achieves better performance in future condition prediction. The results support the hypothesis that shallow machine learning models might have superior results in some conditions compared to deep learning models [18]. Based on previous research, the LSTM model works well on run-to-failure datasets, which is different from the collected dataset [12]. Research comparing deep LSTM, RF, and other machine learning methods implies that LSTM might perform better on a smoother dataset, as the RF performs better RMSE than on a rougher dataset [32]. However, the results shown in Table 1 apply to only the predicted condition of one of the following minutes. Thus, the timespan of predicted output is compared for both RF and LSTM models.

4.2. Accuracy Testing for Different Prediction Timespans

The current model performance reflects the accuracy in predicting one period after the training data. In real-world applications, maintenance personnel often need to assess predictions for longer periods than a single timestep. To address this, the model is tested across different output prediction timespans set to 1, 5, 10, 20, 40, and 60-minute intervals. This allows for an assessment of the model's performance over varying prediction horizons. To manage computational limitations, this evaluation is conducted on the first 480 minutes of training data. The results show that R2 consistently gives negative values for all tests, indicating poor model performance. However, negative R2 values are limited by their inability to quantify the extent of the model's poor performance, as the negative limit of R2 is infinite [29]. This suggests that there might be minimal variation in actual values during the prediction timespan compared to the error between predicted and actual values. Therefore, R2 is not used for prediction timespan comparison in this study. Instead, other performance metrics are utilized to assess the model's accuracy for different prediction timespans. Root Mean Squared Error (RMSE) was selected as the primary metric for multi-horizon analysis due to its greater penalization of large deviations [30], vital in maintenance contexts where significant prediction errors can precipitate unplanned equipment failures and its retention of the same units as the target variable, enabling intuitive comparisons across different forecast horizons. Table 2 shows the RMSE values obtained for both RF and LSTM models across six forecast horizons (1, 5, 10, 20, 40, and 60 minutes) for the Main Drive (MD), Separator (SR), and Fan (FN) Motors, thereby illustrating the relative error growth patterns of each algorithm over increasingly distant prediction intervals.

Table 2. Multi-Horizon RMSE comparison for RF and LSTM models

Horizon (min)	RF RMSE (MD)	LSTM RMSE (MD)	RF RMSE (SR)	LSTM RMSE (SR)	RF RMSE (FN)	LSTM RMSE (FN)
1	0,0116	0,0224	0,0125	0,0318	0,0116	0,0431
5	0,0266	0,0330	0,0307	0,0437	0,0267	0,0571
10	0,0416	0,0360	0,0487	0,0571	0,0419	0,0751
20	0,0453	0,0431	0,0535	0,0788	0,0451	0,1080
40	0,0529	0,0556	0,0635	0,1046	0,0510	0,1556
60	0,0605	0,0650	0,0739	0,1184	0,0573	0,1864

The RMSE trends illustrate that the RF model maintains a comparatively flat error trajectory across all three motors as forecast horizons extend. For the main drive motor, RF's RMSE increases modestly from 0,0116 at a 1-minute horizon to 0,0605 at 60 minutes, whereas the LSTM model's RMSE escalates more steeply from 0,0224 to 0,0650, reflecting over a two-fold amplification of error. Similarly, RF's error growth for the separator motor (0,0125 to 0,0739) and the fan motor (0,0116 to 0,0573) remains under 60, while LSTM's errors rise to 0,1184 and 0,1864, respectively.

This pronounced divergence confirms RF's inherent robustness to cumulative uncertainty, a consequence of its ensemble averaging and feature bagging mechanisms that mitigate sensor noise and drift. In practical CBM implementation, such stable error bounds enable maintenance planners to define dynamic alarm thresholds with predictable confidence intervals, minimising false positives and missed detections, thereby optimizing maintenance scheduling and reducing unplanned downtime. Mean Absolute Error (MAE), as shown in Table 3, offers an alternative perspective by equally weighting all deviations. RF's MAE for the main drive motor climbs from 0,0116 at one minute to 0,0564 at 60 minutes, whereas LSTM's MAE rises from 0,0224 to 0,0602. For the separator and fan motors, RF MAE peaks at 0,0688 and 0,0532, while LSTM MAE surpasses 0,1130 and 0,1783. The smaller MAE increases of RF underscore its capacity to sustain low average prediction errors over extended horizons. From an operational standpoint, this reliability in average error magnitude translates to tighter maintenance windows: teams can anticipate that forecasts will deviate, on average, by no more than a known threshold, thereby planning resource allocation and spare-parts provisioning with greater precision.

MAPE conveys forecast errors in relative terms, facilitating comparisons across motors with differing scale characteristics. Table 4 shows RF's MAPE remaining under 3,5723 % for the main drive, 3,7830 % for the separator, and 3,5755 % for the fan motor across all forecast horizons.

Table 3. Multi-Horizon MAE comparison for RF and LSTM models

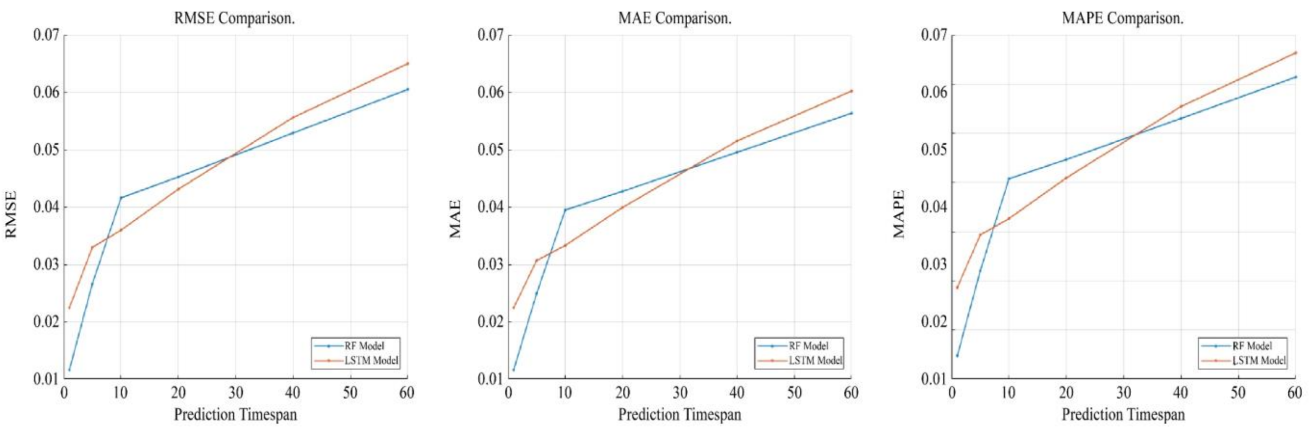
Horizon (min)	RF MAE (MD)	LSTM MAE (MD)	RF MAE (SR)	LSTM MAE (SR)	RF MAE (FN)	LSTM MAE (FN)
1	0,0116	0,0224	0,0125	0,0318	0,0116	0,0431
5	0,0249	0,0307	0,0290	0,0415	0,0251	0,0551
10	0,0395	0,0333	0,0465	0,0538	0,0396	0,0716
20	0,0427	0,0399	0,0506	0,0742	0,0423	0,1020
40	0,0496	0,0515	0,0594	0,0993	0,0476	0,1475
60	0,0564	0,0602	0,0688	0,1130	0,0532	0,1783

In contrast, LSTM's relative errors expand dramatically, exceeding 6 % for the separator and 12 % for the fan motor at 60 minutes. These results emphasize RF's consistent relative accuracy, critical for unified, cross-equipment maintenance policies. In practice, maintenance engineers can set relative degradation thresholds (e.g., a 5 % change) with confidence that RF forecasts will reliably indicate meaningful condition shifts, whereas LSTM's inconsistent relative errors would necessitate motor-specific threshold adjustments and complicate decision protocols.

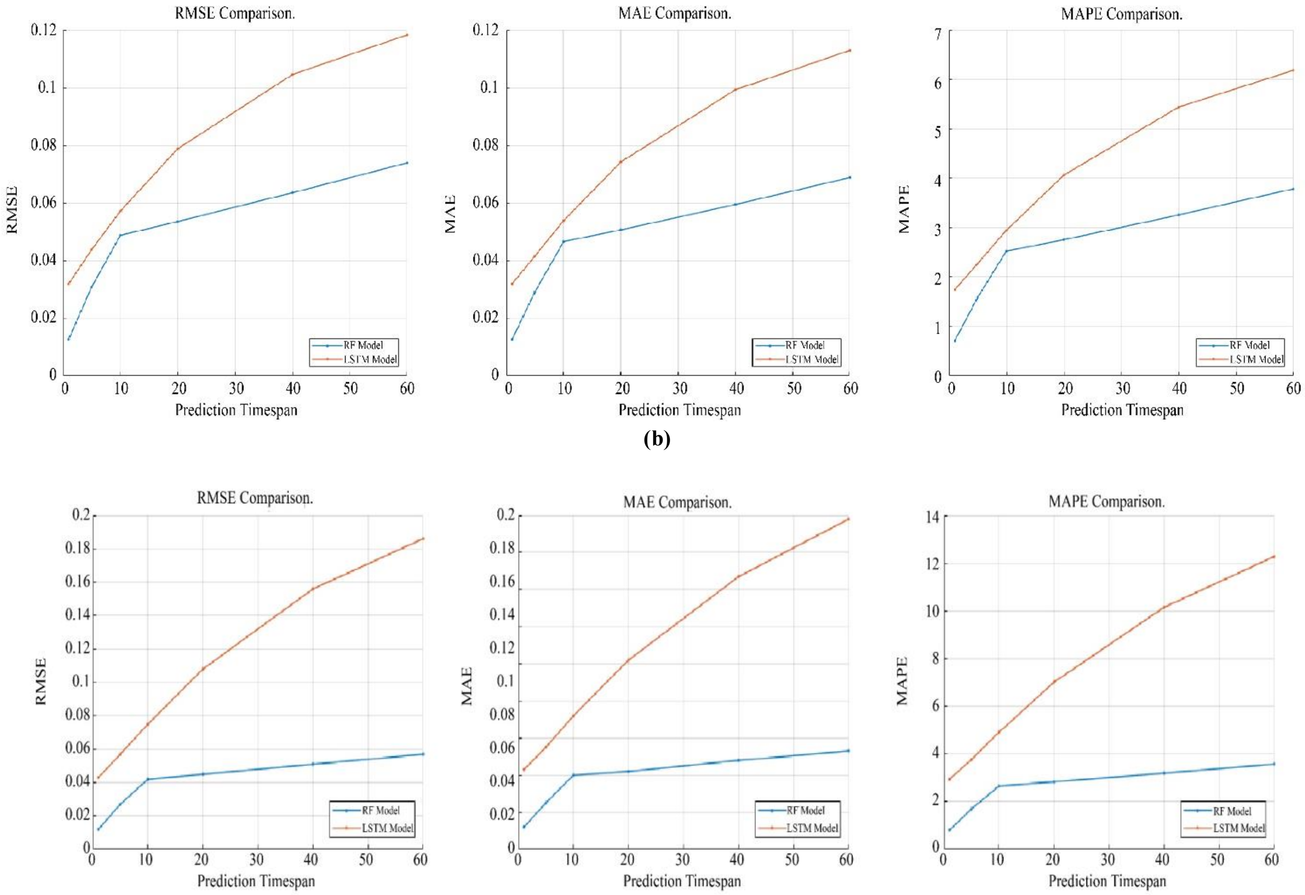
Table 4. Multi-Horizon MAPE comparison for RF and LSTM models

Horizon (min)	RF MAPE (MD)	LSTM MAPE (MD)	RF MAPE (SR)	LSTM MAPE (SR)	RF MAPE (FN)	LSTM MAPE (FN)
1	0,7374	1,4326	0,6994	1,7360	0,8040	2,9412
5	1,5997	1,9670	1,5792	2,2681	1,6934	3,7666
10	2,5388	2,1331	2,5202	2,9415	2,6504	4,9065
20	2,7336	2,5471	2,7499	4,0590	2,8328	7,0087
40	3,1522	3,2735	3,2539	5,4393	3,1955	10,1470
60	3,5723	3,8190	3,7830	6,1870	3,5755	12,2769

The comparison between the RF and LSTM models for the three motors, main drive (a), separator (b), and fan (c), using three metrics (RMSE, MAE, MAPE) as shown in Figure 4, indicates that the RF model generally outperforms the LSTM model across most prediction timespans. The performance trends clearly demonstrate that RF provides significant operational advantages over LSTM across all three motors and forecast horizons. In the key 10 - 20 minute window, RF's modest error growth enables narrow alarm thresholds and minimizes both false positives and missed detections. By contrast, LSTM's rapidly increasing errors would compel wider safety margins, leading to either unnecessary maintenance interventions or unplanned equipment failures. This pattern holds true for the main drive (Figure 4(a)), separator (Figure 4(b)), and fan motors (Figure 4(c)), underscoring RF's suitability as the core prognostic model for CBM. Employing RF forecasts allows maintenance operations to be scheduled with precision, optimizing resource allocation and enhancing overall plant reliability. From a practical perspective, embedding RF within a real-time dashboard equips equipment managers with actionable insights, such as condition thresholds that can be set relative to RF's known error envelope, and multi-horizon forecasts (up to 60 minutes) can be generated in under two seconds on standard industrial hardware, supporting rapid decision cycles. The dashboard's retraining feature further guarantees model relevance as new data accrue, maintaining forecast accuracy over the system's lifecycle.



(a)



(b)

(c)

Fig. 4 Comparison of prediction timespan for (a) Main drive motor, (b) Separator motor, and (c) Fan motor.

Nonetheless, RF's ensemble structure lacks explicit temporal dynamics, limiting its sensitivity to abrupt operational regime shifts such as sudden load changes or emergent faults that evolve on atypical timescales. To overcome this, future research should explore hybrid RF-LSTM architectures wherein RF establishes a stable baseline forecast and LSTM flags rapid deviations as anomaly alerts. Additionally, integrating attention-based or Transformer-style deep models augmented with denoising layers could bolster long-horizon accuracy in the presence of heavy noise and non-stationarity. Coarser time-series aggregation (hourly or daily) should also be evaluated to extend prognostic horizons beyond one hour, enabling strategic, plant-wide maintenance planning.

Recent studies in aerospace prognostics have leveraged Transformer architectures to capture complex temporal dependencies in multivariate sensor streams. For example, [33] propose an improved Transformer with predictive vector angle minimization and feature fusion gates, achieving Remaining Useful Life (RUL) prediction accuracies above 95 % on benchmark turbofan datasets. Similarly, the STAR framework introduces a two-stage attention-based hierarchical Transformer that addresses sensor-wise attention and temporal dynamics, demonstrating significant error reductions over vanilla Transformers [34]. However, these methods focus primarily on deep architectures and often require extensive training data and computational resources that slow real-time use, as well as extensive tuning that can lead to overfitting and poor performance on new equipment or changing conditions.

In contrast, this study shows that a well-tuned RF model, supported by a solid preprocessing pipeline, delivers reliable multi-horizon forecasts using relatively little data and standard hardware. To further improve performance without the full complexity of Transformers, future work could explore hybrid approaches, pairing RF's strength in handling noisy data with lightweight attention modules to catch sudden anomalies. Techniques like data augmentation, transfer learning from other industries, and on-device raining would help maintain model accuracy as operating conditions change.

Based on the performance, RF should serve as the primary prognostic engine for CBM, providing consistent, low-error forecasts that enable proactive maintenance scheduling and efficient resource allocation. Meanwhile, LSTM remains suited to ultra-short-term anomaly alerts (under five minutes), supporting a hybrid framework in which

LSTM handles immediate alarms and RF manages extended prognostics. Deploying this combined approach can minimize unplanned downtime, streamline maintenance cycles, and yield substantial cost savings in industrial settings.

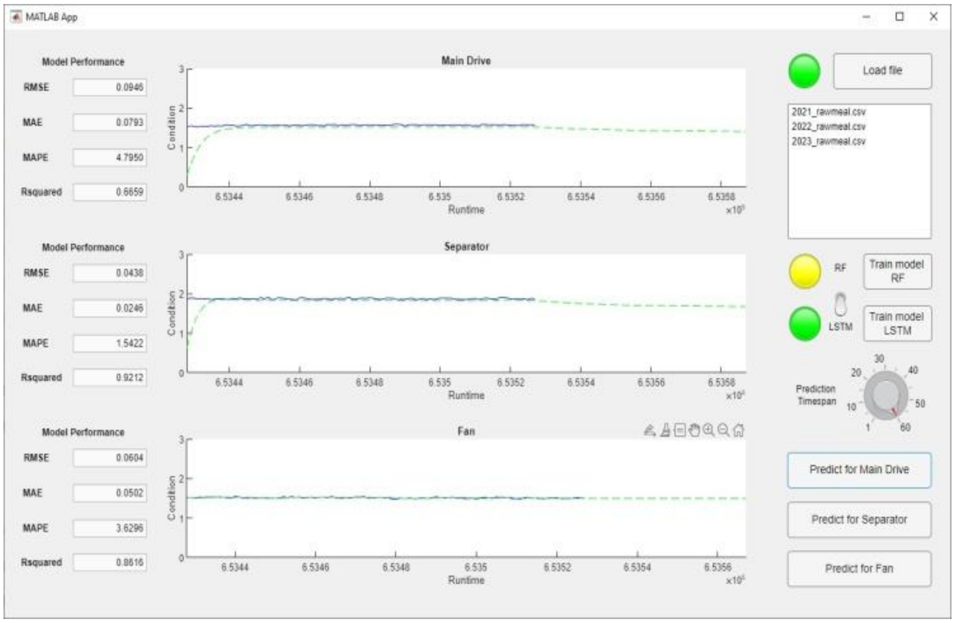
4.3. Dashboard for Visualization

The dashboard visualization is created using MATLAB's app designer. As shown in Figure 5, the dashboard visualization provides a user-friendly interface to visualize the current motor condition, predicted motor condition, and prediction accuracy metrics.

It also supports file uploads for prediction data, allows users to choose between RF and LSTM models, and enables the configuration of the prediction timespan. The app supports file upload for prediction data, the option to choose between RF and LSTM models, and the ability to configure the prediction timespan. Each model features a dedicated readiness indicator, employing a green/yellow/red schema to signal whether the model is uninitialized, on standby, or ready for prediction. This ensures that forecasts are generated only with compatible, up-to-date models, and it allows maintenance personnel to retrain either model within the application as new data becomes available.

A single "Predict" button executes the chosen model for the selected motor (Main Drive, Separator, or Fan) and desired forecast horizon (1–60 minutes), as configured via a convenient knob control. The resulting condition forecasts are overlaid in real time on the motor's actual condition time series, displayed on separate axes for each motor.

Adjacent performance panels report key accuracy metrics computed on the latest 20 % of data, enabling users to assess model reliability at a glance and adjust forecasting horizons or model selection accordingly.


MATLAB App

Model Performance
RMSE 0.0946
MAE 0.0793
MAPE 4.7950
Rsquared 0.6659

Model Performance
RMSE 0.0438
MAE 0.0246
MAPE 1.5422
Rsquared 0.9212

Model Performance
RMSE 0.0604
MAE 0.0502
MAPE 3.6296
Rsquared 0.8616

2021_rawmeal.csv
2022_rawmeal.csv
2023_rawmeal.csv

Fig. 5 Dashboard visualization

The dashboard has been successfully tested on a computer with an Intel Core i7-10700KF CPU, NVIDIA GeForce RTX 3080 GPU, and 32 GB of RAM. It can predict motor conditions for various time spans ranging from 1 to 60, and the performance of each motor's model can be quickly analyzed for accuracy.

The model could be retrained with the latest sensor data to maintain its relevancy, and potential failures can be identified when the predicted condition surpasses the upper threshold. By packaging the app for desktop, web, or standalone deployment, the dashboard could be the solution for complex ML or DL algorithms and actionable maintenance decision-support, providing operators with clear, immediate insights into equipment health and predictive accuracy.

5. Conclusion

This study implements Condition-Based Maintenance (CBM) for a raw mill system by integrating continuous sensor monitoring, predictive analytics, and interactive visualization for three key motors: the main drive, separator, and fan. Through a data-driven framework, Random Forest (RF) and Long Short-Term Memory (LSTM) models were evaluated side by side to determine which approach delivers the most reliable condition forecasts under noisy, high-dimensional operating conditions.

The result shows that the RF model outperforms the LSTM model in nearly all evaluation metrics for all three motors. For the MD, the RF model achieved an RMSE of 0.0235, MAE of 0.0113, MAPE of 0.7356, and R^2 of 0.9790, compared to the LSTM model's RMSE of 0.0479 and R² of 0.9143. Similarly, the RF model demonstrated superior performance across all metrics for the SR and FN, including RMSE values of 0.0237 and 0.0221, respectively, and R² values above 0.9765.

The RF model's consistent performance across all motors and prediction timespans underscores its robustness and suitability for the provided dataset, particularly when compared to the LSTM model. The results align with prior research, which suggests that shallow machine learning models often outperform deep learning models in scenarios where data is limited or noisy.

To enhance practical implementation, a dashboard was developed using MATLAB App Designer, enabling real-time condition monitoring, model training, and performance analysis. This interactive tool supports the selection of models, prediction timespans, and retraining with updated sensor data, ensuring adaptability to evolving operational conditions. Overall, the RF model is recommended for CBM in this application, as it delivers superior predictive accuracy and reliability, making it a valuable tool for industrial maintenance strategies. For future research, exploring alternative deep learning model configurations tailored to various input and prediction timespans is recommended to optimize condition prediction. Additionally, resampling the time series data into hourly, daily, or weekly intervals could enable longer prediction horizons, enhancing the applicability of these models for long-term maintenance strategies. These advancements could further refine predictive accuracy and expand the utility of CBM methodologies in industrial systems.

Funding Statement

This research was funded by the Research Center for Regional Development and Community Empowerment, Institut Teknologi Sepuluh Nopember [2192/PKS/ITS/2023]

Acknowledgments

The authors sincerely thank the industry partner for providing the crucial dataset and resources for this research. Also acknowledge the financial support from Institut Teknologi Sepuluh Nopember for making this work possible.

References

[1] Joao M. Uratani, and Steve Griffiths, “A Forward-Looking Perspective on the Cement and Concrete Industry: Implications of Growth and Development in the Global South,” *Energy Research & Social Science*, vol. 97, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[2] S.C. Nwanya, J.I. Udofia, and O.O. Ajayi, “Optimization of Machine Downtime in the Plastic Manufacturing Industry,” *Cogent Engineering*, vol. 4, no. 1, pp. 1-9, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[3] Seiichi Nakajima, *Introduction to TPM: Total Productive Maintenance*, 1st ed., Productivity Press, Cambridge, MA, 1988. [Google Scholar]

[4] Khashayar Khazraei, and Jochen Deuse, “A Strategic Standpoint on Maintenance Taxonomy,” *Journal of Facilities Management*, vol. 9, no. 2, pp. 96-113, 2011. [CrossRef] [Google Scholar] [Publisher Link]

[5] Deepam Goyal, and B.S. Pabla, “Condition-Based Maintenance of Machine Tools-A Review,” *CIRP Journal of Manufacturing Science and Technology*, vol. 10, pp. 24-35, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[6] Elsayed A. Elsayed, *Reliability Engineering*, 3rd ed., Wiley, Hoboken, 2021. [Publisher Link]

[7] Jeetesh Sharma, Murari Lal Mittal, and Gunjan Soni, “Condition-Based Maintenance using Machine Learning and the Role of Interpretability: A Review,” pp. 1345-1360, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[8] Aparna Gupta, and Chaipal Lawsirirat, “Strategically Optimum Maintenance of Monitoring-Enabled Multi-Component Systems using Continuous-Time Jump Deterioration Models,” *Journal of Quality in Maintenance Engineering*, vol. 12, no. 3, pp. 306-329, 2006. [CrossRef] [Google Scholar] [Publisher Link]

[9] Serkan Ayvaz, and Koray Alpay, “Predictive Maintenance System for Production Lines in Manufacturing: A Machine Learning Approach using IoT Data in Real-Time,” *Expert Systems with Applications*, vol. 173, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[10] Rodney Kizito et al., “The Application of Random Forest to Predictive Maintenance,” *Proceedings of the 2018 IISE Annual Conference*, pp. 354-359, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[11] David Fernández-Barrero et al., “SOPRENE: Assessment of the Spanish Armada’s Predictive Maintenance Tool for Naval Assets,” *Applied Science*, vol. 11, no. 16, pp. 1-18, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[12] Amirhassan Abbasi, Foad Nazari, and C. Nataraj, “Application of Long Short-Term Memory Neural Network to Crack Propagation Prognostics,” *2020 IEEE International Conference on Prognostics and Health Management (ICPHM)*, Detroit, MI, USA, pp. 1-6, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[13] Selin Sunetcioglu, and Taner Arsan, “Predictive Maintenance Analysis for Industries,” *2024 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom)*, Tbilisi, Georgia, pp. 344-347, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[14] Shiv Shankar Sharma, Vivek V, and Ashwini Malviya, “AI-Enhanced Predictive Maintenance in Intelligent Systems for Industries,” *2024 International Conference on Advances in Computing Research on Science Engineering and Technology (ACROSET)*, Indore, India, pp. 1-6, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[15] Nurul Fadzilawati Zainuddin et al., “The Prognostics Approaches and Applications in Aircraft Maintenance Optimization: Review,” *2021 IEEE 12th Control and System Graduate Research Colloquium (ICSGRC)*, Shah Alam, Malaysia, pp. 201-205, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[16] Amit Kumar Jain et al., “A Comprehensive Framework from Real-Time Prognostics to Maintenance Decisions,” *IET Collaborative Intelligent Manufacturing*, vol. 3, no. 2, pp. 175-183, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[17] Song Mao, Xiaofeng Li, and Boyang Zhao, “Remaining Useful Life Prediction based on Time-Series Features and Conformalized Quantile Regression,” *Measurement Science and Technology*, vol. 35, no. 12, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[18] Christian Janiesch, Patrick Zschech, and Kai Heinrich, “Machine Learning and Deep Learning,” *Electron Markets*, vol. 31, no. 3, pp. 685-695, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[19] Md Manjurul Ahsan et al., “Effect of Data Scaling Methods on Machine Learning Algorithms and Model Performance,” *Technologies*, vol. 9, no. 3, pp. 1-17, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[20] Choi W, Kim BHS, “Applying PCA to Deep Learning Forecasting Models for Predicting $PM_{2.5}$,” *Sustainability*, vol. 13, no. 7, pp. 1-30, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[21] Teo Susnjak, Gomathy Suganya Ramaswami, and Anuradha Mathrani, “Learning Analytics Dashboard: A Tool for Providing Actionable Insights to Learners,” *International Journal of Educational Technology in Higher Education*, vol. 19, no. 1, pp. 1-23, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[22] Ali Darejeh, and Dalbir Singh, “A Review on User Interface Design Principles to Increase Software Usability for Users with Less Computer Literacy,” *Journal of Computer Science*, vol. 9, no. 11, pp. 1443-1450, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[23] Ga Young Lee et al., “A Survey on Data Cleaning Methods for Improved Machine Learning Model Performance,” *arXiv Preprint*, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[24] Lattawit Kulanuwat et al., “Anomaly Detection using a Sliding Window Technique and Data Imputation with Machine Learning for Hydrological Time Series,” *Water*, vol. 13, no. 13, pp. 1-20, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[25] Paolo Giordani, “Principal Component Analysis,” *Encyclopedia of Social Network Analysis and Mining*, pp. 1831-1844, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[26] Ian T. Jolliffe, and Jorge Cadima, “Principal Component Analysis: A Review and Recent Developments,” *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, no. 2065, pp. 1-16, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[27] Hassan M. Rabie et al., “Exploring Input Selection for Time Series Forecasting,” *Proceedings of the 2008 International Conference on Data Mining*, Las Vegas, USA, 2008. [Google Scholar]

[28] Philipp Probst, Anne-Laure Boulesteix, and Bernd Bischl, “Tunability: Importance of Hyperparameters of Machine Learning Algorithms,” *Journal of Machine Learning Research*, vol. 20, no. 53, pp. 1-32, 2019. [Google Scholar] [Publisher Link]

[29] Davide Chicco, Matthijs J. Warrens, and Giuseppe Jurman, “The Coefficient of Determination R-Squared is More Informative than SMAPE, MAE, MAPE, MSE and RMSE in Regression Analysis Evaluation,” *PeerJ Computer Science*, vol. 7, pp. 1-24, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[30] Timothy O. Hodson, “Root-Mean-Square Error (RMSE) or Mean Absolute Error (MAE): When to Use Them or Not,” *Geoscientific Model Development Discussions*, vol. 15, no. 14, pp. 5481-5487, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[31] Xue Ying, “An Overview of Overfitting and its Solutions,” *Journal of Physics: Conference Series*, vol. 1168, no. 2, pp. 1-6, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[32] T.T.Q. Nguyen et al., “Comparing High Accurate Regression Models for Short-Term Load Forecasting in Smart Buildings,” *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society*, Singapore, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[33] Zhihao Zhou et al., “An Aircraft Engine Remaining Useful Life Prediction Method based on Predictive Vector Angle Minimization and Feature Fusion Gate Improved Transformer Model,” *Journal of Manufacturing Systems*, vol. 76, pp. 567-584, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[34] Zhengyang Fan, Wanru Li, and Kuo-Chu Chang, “A Two-Stage Attention-Based Hierarchical Transformer for Turbofan Engine Remaining Useful Life Prediction,” *Sensors*, vol. 24, no. 3, pp. 1-19, 2024. [CrossRef] [Publisher Link]