ENHANCED EQUIPMENT RELIABILITY THROUGH PREDICTIVE MAINTENANCE MODELING: A MACHINE LEARNING APPROACH

**Abstract**

The rise of Predictive Maintenance (PDM) moving away from corrective maintenance methods represents a significant advancement in improving operational efficiency and cutting costs effectively. Our paper delves into an examination using cutting edge machine learning approaches. Specifically Long Short Term Memory (LSTM) networks. For implementing PDM and forecasting equipment failures to minimize unexpected production stoppages. We conduct a review of existing literature exploring different models and their application, in various industries underscoring the growing significance of PDM in todays context. Their research approach is quite thorough as they discuss the process of gathering data and preparing it before implementing LSTM networks for forecasting purposes. It appears that LSTM surpasses standard models in terms of effectiveness and precision when anticipating equipment failures lifespan—positioning it as a robust solution, for predictive maintenance challenges. A performance comparison of LSTM to traditional models indicates its greater accuracy and dependability in anticipating equipment failures, establishing it as a viable solution for predictive maintenance difficulties. This study's findings provide a significant addition to the field by presenting a realistic, machine learning-based framework for PdM, proving its potential to reduce costs and improve operational efficiency. Furthermore, this study offers up new areas for future research, such as the investigation of other machine-learning models and the use of real-time data to increase prediction accuracy. The consequences of this study highlight the revolutionary power of sophisticated predictive approaches for improving maintenance tactics and enhancing industrial practices.

**Keywords**

LSTM; PdM; RNN, Convolution LSTM; Bi-LSTM; Predictive Maintenance

# **|** **Introduction**

Maintenance is responsible for a large amount of industrial production expenses, and it is critical to maintaining operational efficiency and equipment lifetime. Effective maintenance techniques may drastically cut costs, eliminate unexpected production stops, and increase the usable life of machines. As a result, the attention of industry specialists has switched to improving maintenance methods, resulting in the development of numerous maintenance techniques. Historically, corrective maintenance (CM), sometimes known as "run-to-failure," was the most common strategy, with repairs or replacements occurring only after equipment breakdown, resulting in costly production downtimes. The enormous costs involved with these unanticipated outages prompted the development of proactive maintenance solutions. Preventative maintenance (PM) arose as a solution, comprising planned inspections and part replacements at regular intervals or after a certain number of operational cycles, as advised by the manufacturer. However, PM presented issues, such as early replacement of working components, which increased maintenance costs, or delayed replacements, which required costly remedial measures. The introduction of Internet of Things (IoT) and Industry 5.0 technologies transformed maintenance processes, resulting in the adoption of condition-based maintenance (CBM). This technique used sensors and monitoring devices to automate inspections, allowing maintenance interventions based on real-time data when equipment performance diverged from established levels. Built on CBM concepts, predictive maintenance (PdM) has arisen as a more complex and successful technique. PdM combines cyber-physical systems (CPS), IoT, robotics, mechatronics, and information technology, and data science to forecast equipment failures and predict the remaining useful life (RUL) of assets, facilitating maintenance planning that minimizes disruptions.

Furthermore, prescriptive maintenance goes beyond PdM by not only forecasting failures but also making proactive recommendations to improve RUL and address particular failure types. For example, although PdM may predict an approaching equipment breakdown owing to increased bearing temperatures, prescriptive maintenance may recommend operating changes, such as decreasing speed, to extend the equipment's life. Despite the advantages of prescriptive maintenance, its successful implementation is contingent upon the robust integration of PdM and CBM, as accurate predictions rely on well-established CBM frameworks. In this framework, the objectives of this study are twofold:

1. To explore the existing methodologies of predictive maintenance
2. Proposed the predictive maintenance solutions in reducing equipment downtime and enhancing productivity in industrial sectors.
3. To evaluate the efficacy of proposed method of machine learning for predictive maintenance of equipment’s breakdown in advance.

Further, section 2 discusses the literature studies of recent work, and Section 3 discusses proposed method. Section 4 shows the result of the performance parameter. Formerly, Section 5 has a result discussion. Thereafter, Section 6 concludes this article.

# **| Literature Review**

***2.1 Overview of Predictive Maintenance Strategies***

Through the use of predictive maintenance, consumers can receive notifications about when their car will need its next scheduled maintenance, when any parts will need to be repaired when warranties or guarantees expire, when engine oils need to be changed, when wheels need to be rotated, and other important details. Therefore, a driver or user's vehicle will always be maintained in better condition if they follow the predictive maintenance schedule rather than using the vehicle carelessly and ignoring it. Predictive maintenance can be performed on the vehicle using data gathered from the Internet of Things (IoT) sensors. Using blockchain technology, this can be connected to incentives so the user gains financially by following the predictive maintenance plan and earning discounts or prizes. The manufacturer, insurance provider, and user can all gain from the blockchain-based solution. The insurance provider can impose penalties or rewards depending on the user's compliance or noncompliance with maintenance schedules. Blockchain's distributed ledger, decentralized data governance, and cryptography primitives and protocols will guarantee data immutability, transparency, and security in this system. Automated cars with IoT connectivity come with capabilities that forecast when the car will need maintenance. This might be thought of as a hardware map that is capable of recording all of the system's parameters. The database prompts the user if a parameter crosses an uncontrollable threshold. Furthermore, the system is able to forecast the system's overall health. It is important to remember that Hyundai Motors' Bluelink technology records every system parameter.

Everything that occurs throughout the car's whole history-both good and bad is tracked. The database also keeps track of any instances in which the motorist exceeded the speed limit or disregarded any safety instructions. Not only is it recorded, but it also indicates what needs to be done next, allowing for predictive maintenance. Chip-based predictive maintenance system gathers vehicle data and is cloud-connected. After processing, the data can be used to predict which portion needs maintenance. Predictive maintenance has become more intelligent with the advent of artificial intelligence (AI). It has the ability to forecast a driver's driving proficiency as well.

***2.2 Existing Studies on Predictive Maintenance and Machine Learning***

One of the main components of I4.0 is processing sensor data to facilitate better decision-making [9]. However, the employment of decision-making algorithms is hampered by the uncertainty included in predictive analytics, the torturous process, and the deadlines for making a conclusion. As PdM has become more popular and effective in current centuries, there has been a spike in the development of techniques that can improve the guiding maintenance decisions to increase machine lifetime.

Ansari et al. [10] have proposed an integrated strategy that combines the Dynamic Bayesian Network (DBN) and data-model to forecast future events, identify cause-and-effect correlations, and offer improvements to maintenance planning. A novel architecture for a PHM technique was put forth by Sarazin et al. [11] in an attempt to increase data value extraction. This lambda architecture consists of two levels: a speed layer then a loading layer. The speed layer may apply maintenance guidelines and machine learning algorithm findings via the storage layer. Moreover, the system has to handle a variety of data, which means it has to handle big data issues while maintaining system compatibility. Cheng et al. [12] used advanced technology to develop a predictive maintenance schedule. In order to provide a better equipment’s maintenance approach for structuring amenities, an integrated PdM planning structure based on building information modeling and the potential of the Internet of Thing for facilitate maintenance supervision was developed. This is because both BIM and IoT have the ability to increase the efficiency of FMM. The future state of MEP components has been predicted using the ANN and SVM machine learning approaches. Calabrese et al. [13] developed a strategy for a large Italian woodworking company that uses machine learning on industrial woodworking machinery. Predicted failure probability is calculated from time series event data using tree-based categorization models. To estimate the residual useful lifetime (RUL) of equipment used in the timber industry, researchers employed temporal feature engineering approaches and trained a set of classification algorithms. A self-adaptable sample of machineries is checked without turning-off the machine’s working to show the efficacy of the suggested technique. Uhlmann et al. [14] proposed an ML model-based method for exploring and visualizing offline data from various sources. In this instance, clusters were found using data from three sensors. They established conventional machines tool’s operation along with three incorrect scenarios. These results were also applied to the creation of a ailment-screening model, that facilitated the creation of machines tool for PdM solutions.

In order to reduce the firmness of recurrent neural networks (RNN) algorithm, Markiewicz et al. [15] created a unique method in which the processing is handled by sensors. A sensor conducted the data analysis, and at that point, only one packet was transmitted, perhaps due to the machine operating wrongly. Since, the ultra-low-power electronics used in the arrangement, sensors can be powered by collected energy. This configuration significantly enhanced processing power with little energy use. A random forest (RF)-based upon deep learning technique was presented by Zenisek et al. [16] to detect potential concept drift behavior in a continuous data flow. The concept was to use sensor-equipped equipment to provide data on a specific equipment state so that future failures and deterioration might be identified. These developments demonstrated the possibility of reducing material and time costs by removing failures and enhancing performance. However, screening high-quality data was required to develop computer models. Additionally, based on earlier synthetic dataset experiments, a novel technique was devised to determine the implications of data as a potential indication of atypical system performance.

The application of PdM for a milling cutting tool explained by Traini et al. [17] and verified with real datasets. Overall, this model provided a basic framework for creating a wear level analysis tool, a basic engineering tool, and failure avoidance to increase productivity in conjunction with machine-human collaboration. Chen et al. [18] developed the Cox proportional hazard deep learning (CoxPHDL) technique to solve the problem of restricted and cleansed data issues during maintenance. In order to produce a usable result, the primary goal is to consider the assistances of enhanced dependability and machine learning. Using an autoencoder, a valid representation of nominal data was first produced. Next, the CoxPHM was used to calculate the censored data's time-between-failures (TBF). A BIM and IoT-based FMM model presented by Cheng et al. [19]. This paradigm included application and information layers to give the best maintenance performance. SVM and ANN machine learning approaches were used to evaluate the mechanical, electrical, and piping components' likely condition. In order to perform PdM on nuclear infrastructure, a machine learning technique was developed by Abidi et al. [20]. The uncommon events that might arise in a nuclear structure were compared and navigated using a logistic regression (LR) and an SVM. The best accurate assessment results were produced by the SVM. Furthermore, the parameters of the LR and SVM algorithms were optimized. A novel model with a significantly inferior possibility density was developed to correlate data from the nuclear infrastructure, albeit the prior research used a large amount of data.

Abidi et al. [20] introduced a five-phase intelligent PdM planning outline, that encompasses statistics correction, data normalization, feature collection, accurate decision-making, and evaluation. Predictive maintenance increases manufacturing sustainability by reducing breakdowns, failures, and material waste. Time and material waste can both be decreased with the use of an efficient PdM. For preventive maintenance, Mishra and Manjhi [21] have suggested a failure prediction model. The proposed machine learning approach makes use of many inputs to forecast the future breakdown times of ATMs at the module level. The PdM model with NN estimator was given by Rahhal and Dia [22]. A big database was created utilizing the data that was gathered from various IOT sensors. Additionally, the prediction made using the gathered data has produced encouraging results. Vanilla-RNN and LSTM-RNN, two distinct NN types, have been applied, and these two have produced superior forecasts for lights of bulb.

Additionally, M. Paolanti et al. [23] worked on the PdM and have provided a Random Forest machine learning technique. Using a variety of sensors and communication protocols, real-time industrial data is gathered, and examined on the Azure’s Cloud architecture, and the machine learning model is then used. Lastly, a comparison of the outcomes utilizing the simulation tool analysis has also been done. There have been three categories of historic data utilized. Zhang et al. [24] proposed two-layer method for predicting manufacturing line breakdown. Methods like PCA and grouping have been applied. Additionally, Random Forest is used to deploy the final model, that performs better. To overcome the shortcomings of existing research works, such as lengthier training times, high processor complexity, and the time-data dependency when using statistical analysis methods, a new model that outperforms previous attempts must be developed. When industrial equipment breaks down, there is a significant loss of output and quality, as well as unscheduled downtime until the unit is able to function as intended. Therefore, PdM is anticipated to decrease equipment production line unscheduled downtime. PdM also helps to reduce the amount of time and money needed for unnecessary maintenance as well as repairs if a machine breaks down. PdM thus guarantees the best possible use of resources. Thus, PdM can be carried out by the use of mechanical analysis, lubricant analysis, electromagnetic thermography, acoustic monitoring, and equipment vibration analysis. The car part production machine is one of the various pieces of machinery used in manufacturing industries that is being studied further to meet the goals of the research. The development of PdM systems saves money, time, and prevents the entire manufacturing line from breaking down. The paper by Li et al. [14] categorized potential system failures into four groups: atmospheric influence, element letdown, human mistake, and incorrect treatment technique.

***2.3 Industry Trends and Applications***

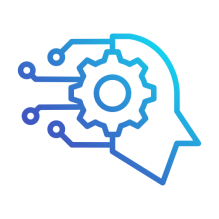
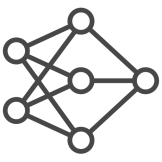
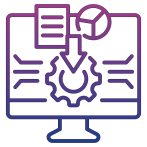
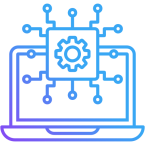
Industry 4.0 makes a number of modifications to the industrial automation paradigm. The notion of intelligent manufacturing can be implemented through the use of Internet of Things and CPS technology, that introduce cognitive automation and ultimately lead to the creation of smart products and services [25]. Businesses must deal with the problems of a considerably more dynamic environment as a result of this new technique. Most of these industries are non-spontaneous to handle this trending reality, in which having a lot of events do not always render into improved output [26].

One of the tenets of Industry 4.0, that converts conventional production into intelligent, sensor-embedded factories with pervasive technologies, are the source of the massive volume of data. A practical implementation of this idea involves employing data analytics in the development of Decision Support Systems, which can facilitate more effective decision-making and hasten the recovery from failures [27], [28]. The Design Support System in Industry 5.0, that evolved from the Decision Support Systems, offers strategy supports by utilizing machine learning algorithms to suggest, for instance, new product versions based on a product's attributes [29]. PdM is another application that makes extensive use of data; in this case, CPS’s can offer self-alertness and self-maintenance intellect. By using this method, the industry is able to forecast the deterioration of product performance and manage and enhance product service requirements on its own [30], [31].

Predictive maintenance has numerous advantages in production settings, nonetheless, it also has downsides that must be resolved. PdM has several advantages, such as enhanced productivity, less system faults [32–35], fewer unscheduled downtimes [36–37], better financial and human resource management [38], and better maintenance intervention planning. In addition to being used to diagnose problems, machine learning (ML) can also be used to prognosticate and anticipate failures. For instance, ML can be used to estimate a machine's lifetime by processing the huge amount of data to train an algorithm.

# **| Methodology**

The methodology used for performance analyses of dataset is shown in Figure 1. Generally, it contains four steps i.e. Data Collection and preprocessing, Training the ML model, Implementation of LSTM, and Performance Evaluation. Further, it is described below as per the methodology used for PdM:



Data Collection

Data Preprocessing

Selection and Training

Implement LSTM

Outcome/Trained Model

Fig. 1 Proposed Methodology

*i. Data Collection and Preprocessing*

We take two datasets from Kaggle to check the effectiveness of the proposed algorithm. The two datasets are 3W and Wind turbine PMSG datasets. The first dataset is on Undesirable events in oil wells and the second one is on Short-Circuit Faults in wind turbines. Before processing, first, we normalized the data between 0 and 1 and removed noise and missing values from data points.

*ii. Selection and Training of Machine Learning Models*

As you seen above, there are many machine learning model in the literature which can deal with this type of problems. So, from those models which model is perfect to deal with this problem is the challenge. As LSTM is a popular machine learning model and it can deal with time series prediction data very properly, we use that model in this scenario. Furthermore, the challenge is how many data points we can use for training; how many data points are used for validation, and how many data points we can use for testing. Here, in two datasets use 80% data for training, 10% for validation, and rest data points for testing.

*iii. Implementation of LSTM*

After validation of data and selection training LSTM model is implemented for Downtime Prediction. In this model, the first layer is an LSTM layer with 100 units followed by another LSTM layer with 50 units. Dropout is also applied after each LSTM layer to control overfitting. Final layer is a Dense output layer with nine unit for 3W dataset and three unit for wind turbine PMSG dataset. Figure 2 shows the architectural diagram of LSTM model and working. The core components of an LSTM cell include the cell state and three gates: the input gate, forget gate, and output gate. The cell state acts as a memory that carries essential information through time steps, while the gates regulate the flow of information. The aforementioned gates use sigmoid and activation functions to handle the information flow, ensuring that the network can maintain relevant data over long sequences. This unique architecture allows LSTMs to overcome the vanishing gradient problem common in traditional RNNs, making them highly effective for tasks like time series prediction, natural language processing, and other applications requiring the modeling of long-range dependencies in data. The forget gate decides what information should be discarded, the input gate controls the addition of new information to the cell state, and the output gate decides what portion of the cell state should be used to generate the hidden state for the next time step.

A diagram of a computer process

Description automatically generated

Fig. 2. LSTM Model Architecture

*iii. Implementation of LSTM*

The output after implementing the proposed model significantly gives the improved prediction of fault of machines that further can be used for predictive maintenance of the industrial equipment in a better way to enhance the productions. Also, it helps to reduce the cost of maintenance that is conventionally done by the mechanics manually, that takes time to find the faults. The increased maintenance time affects production.

***3.1 Description of the Dataset and Features Used***

In this study, two datasets from Kaggle are used to evaluate the efficiency of the suggested technique. These are the 3W and Wind Turbine PMSG datasets [39], [40]. The first dataset is for undesirable occurrences in oil wells, whereas the second is for short-circuit faults in wind turbines. In the 3W and Wind turbine PMSG datasets, all characteristics are utilized. The Wind Turbine PMSG Short Circuit Fault Dataset and the 3W Dataset include detailed time-series data that is required for creating predictive maintenance models in industrial settings. The Wind Turbine PMSG dataset is designed to detect and anticipate short-circuit defects in wind turbine generators, and includes essential electrical and mechanical characteristics such as voltage, current, power production, vibration, and temperature. The 3W dataset, on the other hand, is intended for the oil and gas sector. It collects critical operational data from oil wells such as pressure, temperature, flow rate, vibration levels, and pump speed. Both datasets include a binary target variable (Fault Indicator for the Wind Turbine PMSG dataset and Equipment Status for the 3W dataset) that indicates if a fault has occurred, making them ideal for training machine learning models to predict equipment failures.

**Table 1** Sample data of both datasets

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Feature Name** | **Description** | **Data Type** | **Range/Values** | **Dataset** |
| Timestamp | Time of data recording | Datetime | N/A | Both |
| Voltage | Voltage level of the wind turbine generator | Float | Varies | Wind Turbine PMSG |
| Current | Current flow in the generator | Float | Varies | Wind Turbine PMSG |
| Power | Power output of the generator | Float | Varies | Wind Turbine PMSG |
| Frequency | Electrical frequency output | Float | Varies | Wind Turbine PMSG |
| Vibration | Vibration levels detected | Float | Varies | Both |
| Temperature | Temperature within the system | Float | Varies | Both |
| Fault Indicator | Indicates fault occurrence (1 = Fault, 0 = No Fault) | Integer | 0, 1 | Wind Turbine PMSG |
| Pressure | Pressure level within the oil well | Float | Varies | 3W |
| Flow Rate | Flow rate of oil through the well | Float | Varies | 3W |
| Pump Speed | Speed of the oil well pump | Float | Varies | 3W |
| Equipment Status | Operational status (1 = Normal, 0 = Fault) | Integer | 0, 1 | 3W |

These datasets are characterized by their rich and varied continuous features, sample of data is shown in Table 1. ML model datasets are divided into training, validation, and testing sets, with 80% of the data assigned for training, 10% for validation, and 10% for testing. This divide guarantees that the models can effectively learn from a wide range of situations while also being able to generalize to new, previously unknown data. The 50 epochs used for training find a compromise between underfitting and overfitting, allowing the models to converge well while avoiding undue specialization to the training data. Both datasets are critical for enhancing predictive maintenance tactics, allowing models to reliably forecast problems and optimize maintenance schedules. ML models use these datasets to improve operational efficiency, decrease downtime, and prevent costly malfunctions in both the renewable energy sector and the oil and gas industry.

The core objective of predictive maintenance is to forecast equipment breakdowns based on past time series data. Consider a dataset is the binary label indicating whether a fault occurred (1 for fault, 0 for normal operation). The intent is to train a predictive function that estimates the probability of a fault occurring at time . The Long Short-Term Memory (LSTM) network is a form of Recurrent Neural Network (RNN) that excels at collecting temporal relationships in time-series data, making it excellent for predictive maintenance applications. The LSTM cell is characterized by the following equations:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | (Input Gate) | (1) |
|  |  | (Forget Gate) | (2) |
|  |  | (Output Gate) | (3) |
|  |  | (Candidate Cell State) | (4) |
|  |  | (Cell State Update) | (5) |
|  |  | (Hidden State) | (6) |

Where:

* is the input gate, is the forget, is the output gates, and is the candidate cell state.
* , , and are the learnable parameters of the LSTM.
* is the sigmoid activation function, whereas is the hyperbolic tangent activation function.
* denotes element-wise multiplication.

***3.2 Loss Function***

To train the LSTM model, we minimize a loss function that quantifies the difference between the predicted probability and the actual label . For binary classification, the binary cross-entropy loss function is typically used:

(7)

Where indicates the computational parameter (weights and biases of the LSTM layers).

***3.3 Model Training and Optimization***

The Model variables are changed using optimization approaches such as gradient descent. The update rule for the parameters is given by:

(8)

Where:

* + is the rate of learnings.
  + is the value of the loss function relative to the parameter .

***3.4 Evaluation Metrics***

The model's performance is evaluated using metrics like Mean Absolute Error (MAE) and coefficient of predictability . It is calculated as follows:

(9)

(10)

Where:

* is the mean of the actual labels.
* represents the fraction of variation in the dependent variable that can be predicted from the independent variables, with values closer to one suggesting a better fit.

# **| Results**

The metrics used for assessing model performance are Mean Absolute Error (MAE) and R2Score for the goodness of fit of deep learning models. The models underwent training for a total of 50 epochs, a strategic choice made following thoughtful deliberation on the training process. Training a model over multiple epochs offer distinct advantages that contribute to its performance and resilience. The graphs presented in Figure 3(a), 3(b), and 3(c) offer critical insights into the performance of the LSTM model throughout the training process. Each graph represents a key evaluation metric-Loss, MAE, and R2Score-plotted against the number of training epochs. These metrics supported evaluation of model learning on unseen data.

|  |  |
| --- | --- |
|  | |
| Fig. 3(a) Loss vs. Epochs | Fig. 3(b) MAE vs. Epochs |

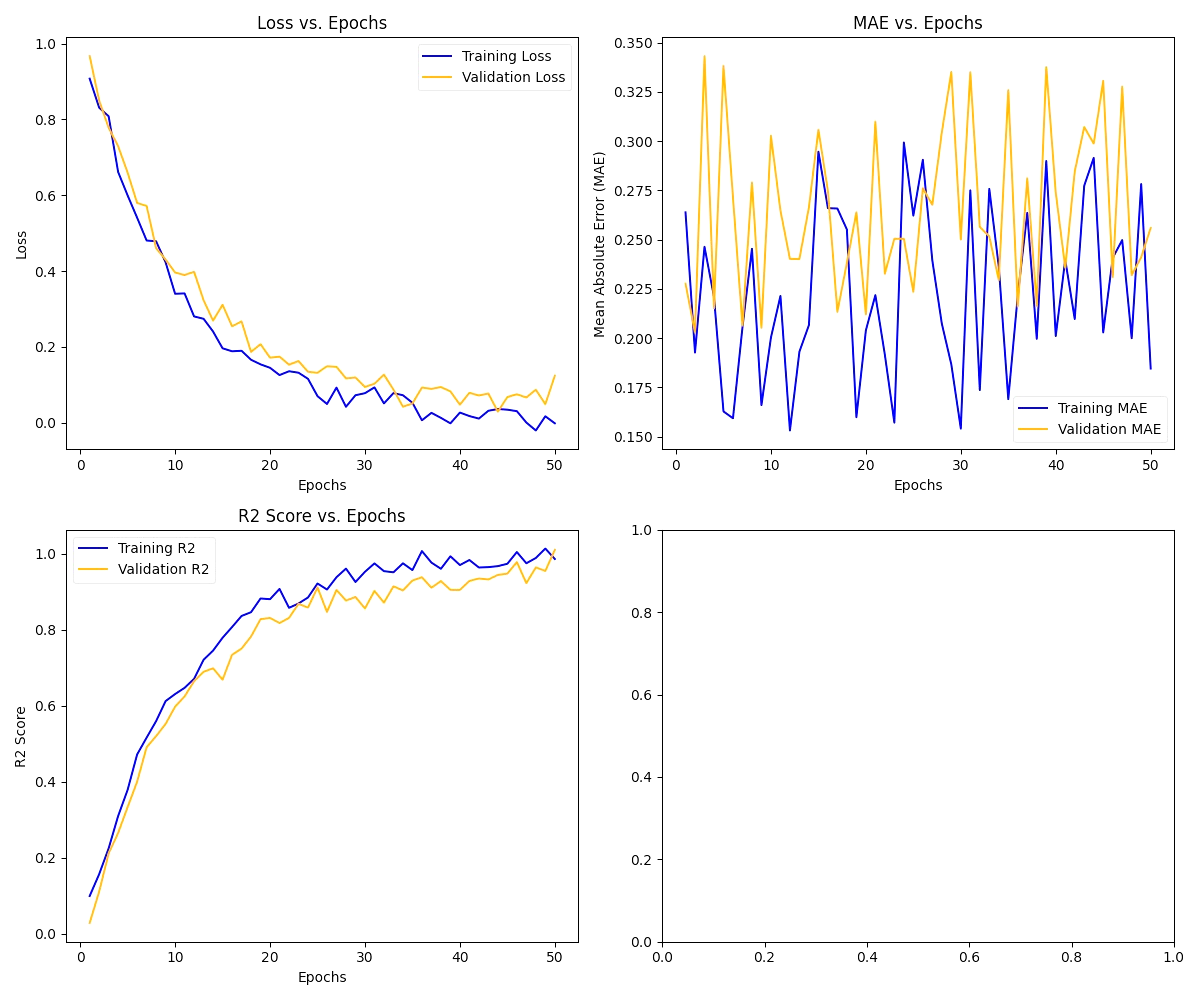


Fig. 3(c) R2Score vs. Epochs

Fig. 3. Performance evaluation of LSTM model

1. ***Loss vs. Epochs***

The first Figure 3(a) shows the Training Loss and Validation Loss across 50 epochs. Loss functions measure how well the model's predictions align with the actual values, with lower values indicating better performance.

* ***Training Loss:*** The training loss curve demonstrates a steady decrease over the epochs, which indicates that the model is progressively learning from the training data. A consistent reduction in loss suggests that the model is fitting the training data more accurately as the training progresses.
* ***Validation Loss:*** The validation loss initially reduces, indicating better generalization to new data. Nevertheless, within a given number of epochs, if the validation loss plateaus or even slightly increases, it may signify the commencement of overfitting, in which the model tends to memorize the training data rather than learning generalizable patterns.

This pattern is common in models that use machine learning and emphasizes the significance of tracking both training and validation loss to find the best stopping point for training.

1. ***MAE vs. Epochs***

The second Figure 3(b) depicts the Mean Absolute Error (MAE) for both the training and validation sets throughout the epochs. MAE is a simple statistic that evaluates the average magnitude of mistakes in forecasts without taking into account their direction. Lower MAE values indicate greater accuracy.

* ***Training MAE:*** The training MAE reduces with time, showing that the model's predictions improve on the training set with learning.
* ***Validation MAE:*** Similar to the training MAE, the validation MAE initially lowers, signaling that the model is better in predicting on unseen data. If the validation MAE remains close to the training MAE, it means that the model is not overfitting and can generalize well to new data.

However, if the validation MAE starts to diverge from the training MAE after several epochs, it may suggest that the model is beginning to overfit the training data, losing its ability to generalize.

1. ***𝑅2Score vs. Epochs***

The third Figure 3(c) presents the 𝑅2Score for both the training and validation sets. The 𝑅2Score, also known as the coefficient of determination, indicates the proportion of variance in the dependent variable that is predictable from the independent variables. An 𝑅2Score closer to 1 indicates a model that explains a high proportion of the variance and thus has better predictive accuracy.

* ***Training 𝑅2Score:*** The training 𝑅2Score increases over the epochs, suggesting that the model is capturing more of the variance in the training data as it learns.
* ***Validation 𝑅2Score:*** It often follows the same pattern as the training 𝑅2Score. A steady improvement in the validation 𝑅2Score implies that the model is growing better at predicting from unseen data. If this score plateaus or begins to decline, it may suggest that the model is starting to overfit.

The collective effect of these three graphs gives a complete picture of the model's training process. The model's capacity to learn and generalize is demonstrated by a consistent decrease in loss and MAE, as well as an improvement in 𝑅2Score. However, careful observation of the validation metrics is crucial, as deviations from the training metrics could signal overfitting. By monitoring these metrics, one can determine the optimal number of training epochs and adjust the model to improve performance. LSTM network's structure, combined with the appropriate loss functions and optimization methods, effectively predicts the equipment failures in time-series data. The decision to allocate 80% of the data for training, 10% for validation, and 10% for testing ensures that the model evaluate well to new data. The selection of 50 epochs for training strikes ensure balance between achieving model convergence and avoiding overfitting, making the model robust for real-world predictive maintenance applications. Further in sub-sections 4.1 and 4.2, we compare the LSTM model with RNN and various versions of LSTM like convolutional LSTM, Bi convolution LSTM. Results are presented dataset-wise to analyze.

***4.1 Results on 3W dataset:***

Figure 4 and Table 2 show the performance of RNN using R2Score and MAE. After getting the results, it has been analyzed that RNN gives the best result at 50 units, and as the units increases the performance get down by increasing MAE.

**Table 2** Results for RNN model on 3W dataset

|  |  |  |
| --- | --- | --- |
| Dynamic Parameters | R2Score | Mean absolute error |
| RNN units = 10 | 0.8273 | 0.0504 |
| RNN units = 50 | 0.8427 | 0.0478 |
| RNN units = 100 | 0.7182 | 0.0657 |

Fig. 4. Performance of RNN model

Figure 5 and Table 3 show the performance of the Convolutional LSTM model on 3Wdataset using R2Score and MAE. After getting the results, it has been analyzed that Convolutional LSTM gives the 0.9362 R2Score at 100 units, and MAE is 0.0213 whereas the R2Score at 50 units is less i.e. 0.9312.

**Table 3** Results for Convolutional LSTM model on 3Wdataset

|  |  |  |
| --- | --- | --- |
| Dynamic Parameters | R2Score | Mean absolute error |
| LSTM units = 10 | 0.936 | 0.0209 |
| LSTM units = 50 | 0.9312 | 0.0234 |
| LSTM units = 100 | 0.9362 | 0.0213 |

Fig. 5. Performance of Convolutional LSTM

Fig. 6. Performance of Conv. Bi-LSTM

Figure 6 and Table 4 show the performance of the Conv. Bi-LSTM model on 3Wdataset using R2Score and MAE. After getting the results, it has been analyzed that Bi-LSTM gives the 0.9847 R2Score at 100 units, and MAE is 0.0227 whereas the R2Score at 10 units is less i.e. 0.9557 and MAE is 0.0475. So, it indicates the MAE decreases as the units increases.

**Table 4** Results for Conv. Bi-LSTM model on 3Wdataset

|  |  |  |
| --- | --- | --- |
| Dynamic Parameters | R2Score | Mean absolute error |
| units = 10 | 0.9557 | 0.0475 |
| units = 50 | 0.985 | 0.0203 |
| units = 100 | 0.9847 | 0.0227 |

***4.2 Results on*** ***Wind turbine PMSG dataset:***

The results for RNN on Wind turbine PMSG dataset is shown in Figure 7 as a graphical form for better understanding. Figure 7 and Table 5 show the performance of the RNN model on Wind turbine PMSG dataset using R2Score and MAE. After getting the results, it has been analyzed that RNN model gives the 0.6173 R2Score at 10 units, and MAE is 0.0688 whereas the R2Score at 100 units is less i.e. 0.5587and MAE is 0.0988, which is higher than other compared units. So, it indicates the MAE increases as the units increases.

**Table 5** Results for RNN model on Wind turbine PMSG dataset

|  |  |  |
| --- | --- | --- |
| Dynamic Parameters | R2Score | Mean absolute error |
| RNN units = 10 | 0.6173 | 0.0688 |
| RNN units = 50 | 0.6458 | 0.0897 |
| RNN units = 100 | 0.5587 | 0.0988 |

Fig. 7. Performance of RNN model

Figure 8 and Table 6 show the performance of the Convolutional LSTM on Wind turbine PMSG dataset on R2Score and MAE. It has been analyzed after getting the results, that Convolutional LSTM gives R2Score is 0.8051at 100 units, and MAE is 0.0329, whereas the R2Score at 10 units is less i.e. 0.7961 and MAE is 0.0305. So, it indicates the MAE get slight change in its value with the increases in units but R2Score increases as the unit increases.

**Table 6** Results for Conv-LSTM model on Wind turbine PMSG dataset

|  |  |  |
| --- | --- | --- |
| Dynamic Parameters | R2Score | Mean absolute error |
| LSTM units = 10 | 0.7961 | 0.0305 |
| LSTM units = 50 | 0.8013 | 0.0388 |
| LSTM units = 100 | 0.8051 | 0.0329 |

Fig. 8. Performance of Convolutional LSTM

Figure 9 and Table 7 show the performance of the Convolutional Bi-LSTM on Wind turbine PMSG dataset using R2Score and MAE. It has been analyzed after getting the results, that Convolutional Bi-LSTM gives R2Score 0.8712 at 100 units, and MAE is 0.0225, whereas the R2Score at 10 units is less i.e. 0.8448 and MAE is 0.0411. So, it indicates that MAE decreases with increase of units whereas R2Score increases as the unit increases.

**Table 7** Results for Conv. Bi-LSTM model on Wind turbine PMSG dataset

|  |  |  |
| --- | --- | --- |
| Dynamic Parameters | R2Score | Mean absolute error |
| units = 10 | 0.8448 | 0.0411 |
| units = 50 | 0.8558 | 0.0236 |
| units = 100 | 0.8712 | 0.0225 |

Fig. 9. Performance of Conv. Bi-LSTM

# **| Discussion**

This section discusses the interpretation of results, model accuracy, and implications of research findings with future directions. Figure 4 to Figure 6 illustrate the R2Score and the Mean absolute error of 3W dataset. Thereafter, Figure 7 to Figure 9 demonstrate R2Score and Mean absolute error of the wind turbine dataset. In this research, RNN and two versions of LSTM are used to compare our method with LSTM. When R2Score is high then the performance of the method is good and when mean absolute error is less than the performance of a method is good. Using Table 2 to Table 9 we can conclude that in both dataset the worse-performing method is RNN, and the best method is Convolutional Bi-LSTM. The performance of Convolutional LSTM is worse than Convolutional Bi-LSTM, and the performance of LSTM is worse than Convolutional LSTM. In RNN, Convolutional LSTM, and in Convolutional Bi-LSTM we use a variable set of units to check the performance of that method. The variable units are 10, 50, and 100 respectively. In most cases using 100 units the performance is the best among all.

*Implications of the Research Findings:* In the research, our methodology shows the effectiveness of LSTM and its various version in two datasets. Here, the LSTM used a fixed set of LSTM-units. In RNN, Convolutional LSTM, Convolutional Bi-LSTM number of units keep varies. From the results it can be illustrated Convolutional Bi-LSTM perform the best among all comparing methods.

*Practical Applications and Future Directions:* In this article, two datasets used to check the effectiveness of the LSTM and its various version. From all LSTM version, Convolutional Bi-LSTM perform the best. In future we can use some other version of LSTM, attention model to check the performances. We can use this type of models where time series dataset is available or where a system produce data within certain time intervals.

# **| Conclusion**

This study presents a comprehensive evaluation of predictive maintenance models, focusing on the performance of Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) variants, including Convolutional LSTM and Convolutional Bi-LSTM. The analysis, based on two critical datasets-3W and wind turbine, demonstrates that the Convolutional Bi-LSTM consistently outperforms other models, with the highest R2Scores and the lowest Mean Absolute Error (MAE) across various unit configurations. The graphs indicate a steady decline in both training and validation losses over the epochs, with validation loss beginning to diverge slightly, suggesting the onset of overfitting. The lower MAE and higher R2Score demonstrate the model's rising accuracy and capacity to grasp temporal relationships in the data. Despite the encouraging results, the model's performance is fundamentally constrained by the quality and amount of accessible data, which may limit its applicability to various industrial situations. Furthermore, while the Convolutional Bi-LSTM has tremendous promise, more advancements might be made by investigating advanced architectures such as attention mechanisms and incorporating real-time data streams to improve flexibility and robustness in dynamic contexts. Finally, this study indicates the efficacy of LSTM-based models, namely Convolutional Bi-LSTM, in predictive maintenance applications, laying the groundwork for further research. The offered approaches have significant promise for larger industrial use, notably in improving maintenance schedules and reducing unnecessary downtime. Future research should focus on overcoming the highlighted constraints and investigating the recommended additions to advance the area of predictive maintenance.

**Availability of Data:** NA

**Conflict of Interest:** The authors have no conflict of interest.

**References**

1. Bevilacqua, M., & Braglia, M. (2000). Analytic hierarchy process applied to maintenance strategy selection. *Reliability Engineering and System Safety, 70*(1), 71–83.
2. Mobley, R. K. (2002). An introduction to predictive maintenance. *Elsevier.* Doi: 10.1016/B978-0-7506-7531-4.X5000-3
3. Hao, Q., Xue, Y., Shen, W., Jones, B., & Zhu, J. (2010). A decision support system for integrating corrective maintenance, preventive maintenance, and condition-based maintenance. In *Construction Research Congress 2010* (pp. 470–479). American Society of Civil Engineers. Doi: 10.1061/41109(373)47
4. Lee, J., Kao, H. A., & Yang, S. (2014). Service innovation and smart analytics for Industry 4.0 and big data environment. *Procedia CIRP, 16*, 3–8. Doi: 10.1016/j.procir.2014.02.001
5. Bagheri, B., Yang, S., Kao, H. A., & Lee, J. (2015). Cyber-physical systems architecture for self-aware machines in industry 4.0 environment. *IFAC-PapersOnLine, 28*, 1622–1627. Doi: 10.1016/j.ifacol.2015.06.318
6. Sreedharan, R., & Unnikrishnan, V. A. (2017). Moving towards Industry 4.0: A systematic review. *International Journal of Pure and Applied Mathematics, 117*(20), 929–936.
7. Fox, H., Pillai, A. C., Friedrich, D., Collu, M., Dawood, T., & Johanning, L. (2022). A review of predictive and prescriptive offshore wind farm operation and maintenance. *Energies, 15*(2), 504. Doi: 10.3390/en15020504
8. Lepenioti, K., Bousdekis, A., Apostolou, D., & Mentzas, G. (2020). Prescriptive analytics: Literature review and research challenges. *International Journal of Information Management, 50*, 57–70. Doi: 10.1016/J.IJINFOMGT.2019.04.003
9. Amruthnath, N., & Gupta, T. (2018). A research study on unsupervised machine learning algorithms for early fault detection in predictive maintenance. In *2018 5th International Conference on Industrial Engineering and Applications (ICIEA)* (pp. 355–361). IEEE.
10. Ansari, F., Glawar, R., & Sihn, W. (2020). Prescriptive maintenance of CPPS by integrating multimodal data with dynamic Bayesian networks. In *Machine Learning for Cyber Physical Systems* (pp. 1–8). Springer Vieweg, Berlin, Heidelberg.
11. Sarazin, A., Truptil, S., Montarnal, A., & Lamothe, J. (2019). Toward information system architecture to support predictive maintenance approach. In *Enterprise Interoperability VIII* (pp. 297–306). Springer, Cham.
12. Cheng, J. C. P., Chen, W., Chen, K., & Wang, Q. (2020). Data-driven predictive maintenance planning framework for MEP components based on BIM and IoT using machine learning algorithms. *Automation in Construction, 112*, 103087.
13. Calabrese, M., Cimmino, M., Fiume, F., Manfrin, M., Romeo, L., Ceccacci, S., Paolanti, M., et al. (2020). SOPHIA: An event-based IoT and machine learning architecture for predictive maintenance in industry 4.0. *Information, 11*(4), 202.
14. Uhlmann, E., Pontes, R. P., Geisert, C., & Hohwieler, E. (2018). Cluster identification of sensor data for predictive maintenance in a selective laser melting machine tool. *Procedia Manufacturing, 24*, 60–65.
15. Markiewicz, M., Wielgosz, M., Bocheński, M., Tabaczyński, W., Konieczny, T., & Kowalczyk, L. (2019). Predictive maintenance of induction motors using ultra-low power wireless sensors and compressed recurrent neural networks. *IEEE Access, 7*, 178891–178902.
16. Zenisek, J., Holzinger, F., & Affenzeller, M. (2019). Machine learning based concept drift detection for predictive maintenance. *Computers & Industrial Engineering, 137*, 106031.
17. Traini, E., Bruno, G., D’Antonio, G., & Lombardi, F. (2019). Machine learning framework for predictive maintenance in milling. *IFAC-PapersOnLine, 52*, 177–182.
18. Chen, C., Liu, Y., Wang, S., Sun, X., Di Cairano-Gilfedder, C., Titmus, S., & Syntetos, A. A. (2020). Predictive maintenance using Cox proportional hazard deep learning. *Advanced Engineering Informatics, 44*, 101054.
19. Gohel, H. A., Upadhyay, H., Lagos, L., Cooper, K., & Sanzetenea, A. (2020). Predictive maintenance architecture development for nuclear infrastructure using machine learning. *Nuclear Engineering and Technology, 52*, 1436–1442.
20. Abidi, M. H., Mohammed, M. K., & Alkhalefah, H. (2022). Predictive maintenance planning for Industry 4.0 using machine learning for sustainable manufacturing. *Sustainability, 14*(6), 3387.
21. Mishra, K., & Manjhi, S. K. (2019). Failure prediction model for predictive maintenance. In *Proceedings of the 7th IEEE International Conference on Cloud Computing in Emerging Markets (CCEM 2018)* (pp. 72–75).
22. Rahhal, J. S., & Abualnadi, D. (2020). IoT based predictive maintenance using LSTM RNN estimator. In *Proceedings of the 2nd International Conference on Electrical, Communication and Computer Engineering (ICECCE 2020)*.
23. Paolanti, M., Romeo, L., Felicetti, A., Mancini, A., Frontoni, E., & Loncarski, J. (2018). Machine learning approach for predictive maintenance in Industry 4.0. In *Proceedings of the 14th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA 2018)*.
24. Zhang, D., Xu, B., & Wood, J. (2016). Predict failures in production lines: A two-stage approach with clustering and supervised learning. In *Proceedings of the 2016 IEEE International Conference on Big Data*.
25. Kunst, R., et al. (2019). Improving devices communication in Industry 4.0 wireless networks. *Engineering Applications of Artificial Intelligence*.
26. Lee, J., et al. (2014). Service innovation and smart analytics for Industry 4.0 and big data environment. *Procedia CIRP*.
27. Romeo, L., et al. (2020). Machine learning-based design support system for the prediction of heterogeneous machine parameters in Industry 4.0. *Expert Systems with Applications*.
28. O’Donovan, P., et al. (2019). A comparison of fog and cloud computing cyber-physical interfaces for Industry 4.0 real-time embedded machine learning engineering applications. *Computers in Industry*.
29. Boyes, H., et al. (2018). The industrial internet of things (IIoT): An analysis framework. *Computers in Industry*.
30. Wang, K., et al. (n.d.). How AI affects the future predictive maintenance: A primer of deep learning.
31. Schmidt, B., et al. (2018). Predictive maintenance of machine tool linear axes: A case from manufacturing industry. *Procedia Manufacturing*.
32. Liu, Z., et al. (n.d.). Industrial AI enabled prognostics for high-speed railway systems.
33. Kitchenham, B., et al. (2010). Systematic literature reviews in software engineering – A tertiary study. *Information and Software Technology*.
34. Zenisek, J., et al. (2018). Streaming synthetic time series for simulated condition monitoring. *IFAC-PapersOnLine*.
35. Bumblauskas, D., et al. (2017). Smart maintenance decision support systems (SMDSS) based on corporate big data analytics. *Expert Systems with Applications*.
36. Nuñez, D. L., et al. (2018). Ontoprog: An ontology-based model for implementing prognostics health management in mechanical machines. *Advanced Engineering Informatics*.
37. Cheng, J. C. P., et al. (2020). Data-driven predictive maintenance planning framework for MEP components based on BIM and IoT using machine learning algorithms. *Automation in Construction, 112*, 103087.
38. Daniyan, I., et al. (2020). Artificial intelligence for predictive maintenance in the railcar learning factories. *Procedia Manufacturing*.
39. Adônis, B. (2020). *Wind turbine PMSG - Short-Circuit Fault*. Kaggle.com. <https://www.kaggle.com/datasets/brunoadonis/wind-turbine-pmsg-short-circuit-fault-mcsa>
40. *3W Dataset - Undesirable events in oil wells*. (n.d.). Kaggle.com. <https://www.kaggle.com/datasets/afrniomelo/3w-dataset>