LSTM-Based Condition Monitoring and Fault Prognostics of Rolling Element Bearings Using Raw Vibrational Data

Yasir Saleem Afridi 1, Laiq Hasan 1, Rehmat Ullah 1, Zahoor Ahmad 2 and Jong-Myon Kim 2,*

1 Department of Computer Systems Engineering, University of Engineering and Technology, Peshawar 25000, Pakistan; yasirsaleem@uetpeshawar.edu.pk (Y.S.A.); laiqhasan@uetpeshawar.edu.pk (L.H.); rehmatullah@uetpeshawar.edu.pk (R.U.)
2 Department of Electrical, Electronic and Computer Engineering, University of Ulsan, Ulsan 44610, Republic of Korea; zahooruou@mail.ulsan.ac.kr

Abstract: The 4.0 industry revolution and the prevailing technological advancements have made industrial units more intricate. These complex electro-mechanical units now aim to improve efficiency and increase reliability. Downtime of such essential units in the current competitive age is unaffordable. The paradigm of fault diagnostics is being shifted from conventional to proactive predictive approaches. As a result, Condition-based Monitoring and prognostics are now essential components of complex industrial systems. This research is focused on developing a fault prognostic system using Long Short-Term Memory for rolling element bearings because they are a critical component of industrial systems and have one of the highest fault frequencies. Compared to other research, feature engineering is minimized by using raw time series sensor data as an input to the model. Our model achieved the lowest root mean square error and outperformed similar research models where time domain, frequency domain, or time-frequency domain features were used as input to the model. Furthermore, using raw vibration data also enabled better generalization of the model. This has been confirmed by evaluating the performance of the developed model against vibration data generated by distinct sources, including hydro and wind power turbines.

Keywords: LSTM; machine learning; prognostics; bearings

1. Introduction

One of the most important components in industrial machines are bearings, which normally operate under a very stressful environment and, hence, are continuously prone to degradation [1]. Thus, bearings are among the few critical and foremost fault-causing components in electro-mechanical systems [2]. When aged, faults like bearing wear and tear and symptoms, such as abnormal vibrations, high temperatures, and misalignments, are introduced into the system. Therefore, a predictive approach toward fault identification is necessary in modern, industrially-revolutionized times to avoid forced downtime.

The existing Condition-based Monitoring (CBM) and system health monitoring approaches are generically classified into (1) physics-based models that require extensive domain knowledge where the models are constructed using mathematical equations, and (2) data-driven models where the training of the model is conducted using historic data generated by the sensors [3,4] because the data-driven models work with historic sensory data collected from the machines. Such data can be recorded online, where the model’s parameters can be updated in real-time. Therefore, this approach makes the data-driven models more attractive for the predictive maintenance of electro-mechanical machines.

Various models have been developed for the effective condition monitoring of bearings, including acoustic emission-based models [5,6]. However, vibration signal analysis [7,8] is one of the most widely used and most effective methods to conduct the prognosis of rotating machines. The occurrence of faults in the rotating machines can result in
machine downtime and is directly related to higher operation and maintenance (O&M) costs, serious accidents, and economic losses [9,10]. The fault diagnosis in rotating machines can therefore be conducted by measuring the speed variations using speed sensors [11,12] or by quantifying the vibration signals using accelerometers [13]. However, fault detection and prognostics using vibration signals are the most widely used approaches. A detailed review of the feature extraction and selection techniques, different classifiers, and deep learning models using the Case Western Reserve University bearing center (CWRU) dataset was conducted [14]. Similarly, a future roadmap for intelligent fault diagnostics has been provided [15], and a systematic review of the recent advancements in mechanical fault diagnostics and prognostics was carried out [16].

Several techniques have been adopted to identify faults in the vibration data, including short-time Fourier transform (STFT) and wavelet transform [17–20]. Among the other adapted approaches are band-pass filtering, phase demodulation, Kalman filters, and deep learning techniques based on deep neural networks [21–25]. A total of 17 different classifiers using the MATLAB Classification Learner toolbox along with support vector machine (SVM), K-nearest neighbors (KNN), and ensemble have been used to evaluate the performance of classifiers for diagnosing the faults in induction motors [26]. Likewise, classical machine learning algorithms have also been used to detect leakages in the waterwall tube of a steam power plant [27].

As computational resources advance, deep learning has been in the spotlight, especially in the area of prognostics, because of its effectiveness in modeling complex systems [28]. Artificial neural networks (ANNs) are the most common deep learning methods used for prognostics because of their outstanding performance against complex non-linear multi-dimensional systems. Their ability to effectively process non-linear information makes them more robust concerning noise. Many ANN architectures, such as Feed-forward, Single and Multilayer Perceptron, Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM), Modular Neural Networks (MNNs), and Convolutional Neural Networks (CNNs), are being used to conduct the CBM and prognostics of industrial and renewable energy systems [29]. To diagnose the bearing faults, a large memory storage retrieval (LAMSTAR) neural network based on an optimized deep learning structure is proposed [30]. A Bayesian deep learning model to characterize the latent structure between RULs and degradation features is used to describe prognostic uncertainties [31]. Based thereon, dynamic decisions pertinent to maintenance and spare-part ordering are taken. The performance of the proposed framework has been validated through comparison with various benchmarking policies based on a C-MAPSS turbofan engine data set. Extensive research on integrating Artificial Intelligence, Big Data, and the Industrial Internet of Things in smart manufacturing and modern dynamic industrial processes and energy generation units under the umbrella of Industry 4.0 is conducted [32]. The issue of data imbalances is solved using virtual sensors to artificially induce different health states in the vibration data [33].

This research, therefore, focused on developing a dynamic and robust prognostic model that can efficiently predict the degradation and faults of the bearings based on the raw vibration data recorded from the sensors. All analyses in this research were conducted in the time domain, and the need for feature engineering was minimized by taking actual sensor data as input. The proposed model was developed to predict the bearing degradation and faults ahead of time. Because the raw vibration data generated by the sensors were used for model training and testing, the model could be effectively generalized. To evaluate the generalization capabilities of the developed model, the model was tested for analyzing the faults in the bearings of both hydro and wind power turbines. It was observed that the model effectively predicted the bearing vibration values irrespective of the data source. Using raw vibration values instead of time-domain features also has an advantage in cases where condition monitoring is carried out online in real-time. Since, at any instant in time, only partial data is available, effectively acquiring time-domain features becomes a difficult ask.
The remaining part of the paper is organized as follows. Section 1.1 discusses the novelty of the research. Section 2 explains the overall methodology, including the justification and architecture of the proposed model. Section 3 explains the process of data acquisition, details about the dataset, and elaborates the experimental setup. Section 3 explains the evaluation metrics and results and also compares the results with those achieved by similar research. The generalization ability of the model is also discussed in Section 3. Finally, the paper is concluded in Section 4.

1.1. Related Research and Novelty

Previous research conducted on bearing vibration datasets mostly focused on the frequency domain [34,35]. At first, the time-series data were converted into the frequency domain, followed by data pre-processing and feature extraction [34]. Such analysis is based on the pretext that the data components with higher frequency are actually the added system noise. Subsequently, low-pass filters were used to remove these higher-frequency components, segregating the system and bearing vibrations. However, in a realistic environment, the vibration signals generated by the machinery (other than the bearings) are an integral part of the data. Segregation of these system vibration signals from the bearing vibration signals requires an extensive domain knowledge and, hence, is highly prone to errors.

Furthermore, in similar research, either time domain features, frequency domain features, or time–frequency domain features, such as mean, standard deviation, kurtosis, and skewness, were considered input(s) to the model [36–41]. However, in cases where prognostics are conducted in real-time, where only partial data generated by the Supervisory Control and Data Acquisition (SCADA) system are available at any time, such statistical feature extraction cannot be carried out effectively.

Therefore, this research concentrated on conducting the model training and testing using the actual raw sensor data as an input to the model. This approach helped reduce the need for extensive feature engineering and domain knowledge and also improved the performance of the model in terms of root mean square error (RMSE). Furthermore, because the model is trained using raw sensor data, it can be effectively generalized. An in-depth analysis of the results and generalization capabilities, along with comparing the results with other similar models, are given in later sections.

2. Methodology

The model was developed and tested using Python programming language. The prediction of the vibration values was carried out by first acquiring the sensory data containing raw vibration values recorded by a test rig. Subsequently, this raw data was pre-processed for the removal of any outliers. The pre-processed data were normalized and then used to train and test a prognostic model based on a machine-learning algorithm. Finally, hyper-parameter testing and fine-tuning of the model were conducted. In the end, the generalization capability of the model was verified by testing it against real vibration data generated by different sources.

Figure 1 shows the flow chart of the methodology, comprising the experimental steps.

2.1. Model Selection

The vibration signals generated by the installed sensors represent the time-series data expressed sequentially. One of the limitations of conventional models based on multi-domain feature extraction, including kurtosis, spectral skewness, and wavelet coefficients, is their ineffectiveness to model the inherent sequential characteristics of the sensory data. Moreover, selecting features for these models requires extensive domain knowledge and feature engineering skills.

Furthermore, the sequential models, including the traditional ANNs, hidden Markov models (HMMs), Kalman filters, and conditional random fields, despite having their ability to handle the sequential data, are incapable of addressing the long-term dependencies. As
in machine condition monitoring based on sensory data, many noisy or non-discriminative signals may exist between two consecutive informative or discriminative signals. Hence, a long delay on a time scale is induced between important data points. This leads to reduced efficiency and performance of the aforementioned models when working with the time series data.

Figure 1. Steps of the methodology.

To address the issues of long-term data dependency, the use of RNNs for handling sequential data has increased substantially [42]. However, one of the shortcomings of RNNs is the problem of gradient exploding and vanishing. Although RNNs can store the previous inputs in the network and can be trained using backpropagation, their ability to cater to long-term dependencies in sequential data is reduced because of the gradient vanishing problem.

Consequently, the LSTM algorithm was introduced [43]. This algorithm prevented the gradient from vanishing or exploding; however, it also addressed the long-term data dependencies by introducing forget gates in the architecture. LSTMs can simultaneously carry out representative learning and model training without additional domain knowledge. Regarding the unprecedented performance against time-series data, the LSTM algorithm was used to develop the prognostic model.

2.2. Long Short-Term Memory (LSTM)

LSTM was first presented in 1997 by Hochreiter and Schmidhuber [43]. It is a type of RNN that can effectively establish a correlation between a priori information and the current state based on a time series. The proposed LSTM network comprises three basic
components, the Forget gate, Input gate, and Output gate, as shown in Figure 2. These components are used to (1) Forget irrelevant information, (2) Add/Update new information, and (3) Pass on the updated information.

where:
- \( C_{t-1} \) represents the cell state of the previous timestamp;
- \( H_{t-1} \) represents the hidden state of the previous timestamp;
- \( C_t \) represents the current cell state;
- \( H_t \) represents the current hidden state.

**Figure 2.** Basic layout of the proposed LSTM network.

The cell state carries the information, in our case the vibration data points along with all timestamps, and is known as long-term memory. The hidden state is known as short-term memory, hence the name Long Short-Term Memory.

Along with some minor linear interactions, the cell state \( C_t \) carries the data points straight throughout the entire chain, acting as a conveyor belt. The addition of significant data points and the removal of less significant data points to the cell state is regulated by the gates depicted in Figure 2. These gates comprise a sigmoid \( \sigma \) neural network layer and a point-wise multiplication operation, as depicted in Figure 3. The model training is conducted in accordance with these nodes.

**Figure 3.** Detailed structure of the proposed LSTM network.

In the first step of training the model, the Forget gate decides whether to keep the previous data points with the time step or to forget them. This process is governed by Equation (1).

\[
f_t = \sigma \left( w_{fx} x_t + w_{fh} h_{t-1} \right) \tag{1}
\]

where:
- \( x_t \) represents the input to the current timestamp;
- \( w_{fx} \) represents the weight matrix associated with the input;
- \( h_{t-1} \) represents the hidden state of the previous timestamp;
- \( w_{fh} \) represents the weight matrix associated with the hidden state.
A sigmoid function is applied to change the value of $f_t$ to between 0 and 1. The cell state of the previous timestamp is then multiplied with $f_t$ to determine how much information to forget or keep using Equations (2) and (3).

$$f_t \times C_{t-1} = 0 \text{ (if } f_t = 0)$$

$$f_t \times C_{t-1} = C_{t-1} \text{ (if } f_t = 1)$$ (3)

The network will forget the cell state of the previous timestamp if the value of $f_t$ is 0 and will retain it if the value is 1.

In the second step of the training, the input gate is used for quantifying the data points, and which new data points are stored in the cell state is decided. This step has two parts. First, the sigmoid input gate layer decides what values are updated. Second, the tanh layer creates a vector of the new values, $N_t$, to be added to the state. The tanh activation function transforms the values to between $-1$ and 1. The process is conducted using Equations (4) and (5).

$$i_t = \sigma(w_{ix}x_t + w_{ih}h_{t-1})$$

$$N_t = tanh(w_{Nx}x_t + w_{Nh}h_{t-1})$$ (5)

Both of these steps are then integrated to update the cell state of the network using Equation (6).

$$C_t = f_t \times C_{t-1} + i_t \times N_t$$ (6)

Based on the value of $N_t$ in the above equation, either the information is added or subtracted from the cell state. For a negative value of $N_t$, the input data points are subtracted from the cell state; if the value of $N_t$ is positive, the input information is added to the cell state.

In the final step of the training, it is decided what values of the cell state are going to the output. The output is based on a filtered version of the cell state. First, a sigmoid activation function is applied to the cell state, as shown in Equation (7).

$$O_t = \sigma(w_{Ox}x_t + w_{Oh}h_{t-1})$$

Second, a tanh activation function is multiplied with the output of the sigmoid layer using Equation (8) to calculate the current hidden state of the network and decide the relevant points to be sent as an output.

$$h_t = O_t \times tanh(C_t)$$ (8)

The current hidden layer $h_t$ in Equation (9) is a function of the long-term memory $C_t$ and the current output $O_t$.

3. Experiments and Results

This section provides a detailed discussion of the dataset used, the experiments conducted, and the results.

3.1. Dataset

In this research study, we used the dataset from the Prognostic Data Repository of NASA, which was made publicly available by the Center of Intelligent Maintenance System (IMS), University of Cincinnati [44]. Details of the data acquired from the IMS database are in Table 1.
Table 1. Details of the dataset.

<table>
<thead>
<tr>
<th>Type of Bearings</th>
<th>Double Rows Rexnord ZA-2115</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. of Bearings</td>
<td>Four (04)</td>
</tr>
<tr>
<td>Shaft Load</td>
<td>6000 lbs</td>
</tr>
<tr>
<td>Shaft Rotational Speed</td>
<td>2000 rpm</td>
</tr>
<tr>
<td>Type of Accelerometers</td>
<td>High Sensitivity Quartz ICP</td>
</tr>
<tr>
<td>No. of Accelerometers (Test 01)</td>
<td>Two (02) Accelerometers on x-axis and y-axis</td>
</tr>
<tr>
<td>No. of Accelerometers (Test 02 &amp; Test 03)</td>
<td>One (01) Accelerometer</td>
</tr>
<tr>
<td>Sampling Rate</td>
<td>20 kHz</td>
</tr>
</tbody>
</table>

Figure 4 depicts the layout of the test rig. The data were collected for 1 s every 10 min. Each file has 20,480 data points, and the name of the file indicates the time at which the data was recorded. The test rig was lubricated using an oil circulation mechanism. The extent of debris stuck to a magnetic switch installed in the oil feedback pipe of the lubrication system indicated bearing degradation.

Figure 4. Test rig by IMS [44].

Figure 5a depicts Bearing 3 with an inner race fault at the end of the experiment recorded as Dataset 1. Figure 5b shows Bearing 1 with an outer race fault at the end of the experiment recorded as Dataset 2.

The plot of vibrational signals of bearing 1 in dataset 02 for the entire run-to-failure-experiment can be seen in Figure 6a. Whereas the plot of vibrational signals of bearing 3 in dataset 03 for the entire run-to-failure-experiment can be observed in Figure 6b. In Figure 6, the x-axis represents the time taken till the occurrence of a fault in the bearings, and the y-axis represents the vibration values recorded by the accelerometers in m/s².

Figure 5. Physical condition of bearings after the run-to-failure experiment [45].

Figure 6a depicts Bearing 3 with an inner race fault at the end of the experiment recorded as Dataset 1. Figure 6b shows Bearing 1 with an outer race fault at the end of the experiment recorded as Dataset 2.

The plot of vibrational signals of bearing 1 in dataset 02 for the entire run-to-failure-experiment can be seen in Figure 6a. Whereas the plot of vibrational signals of bearing 3 in dataset 03 for the entire run-to-failure-experiment can be observed in Figure 6b. In Figure 6, the x-axis represents the time taken till the occurrence of a fault in the bearings, and the y-axis represents the vibration values recorded by the accelerometers in m/s².
3.2. Experimental Setup

The dataset provided by IMS comprises three (03) datasets, each describing an independent run-to-failure experiment. Dataset 02, with a total of 984 files, was used for the training and testing of the model. The dataset was split into training and testing sets. 70\% of the data was used for training the model, and 30\% was used for testing. The results were further verified by testing the trained model using Dataset 03 with 4448 files.

At the end of run-to-failure experiments, in both Dataset 02 and Dataset 03, Bearing 1 and Bearing 3, respectively, developed an outer race fault. Therefore, the results
were verified against Dataset 03 after performing the training and testing of the model on Dataset 02.

3.2.1. Data Pre-Processing

The data is first pre-processed to remove any outliers. After the data cleansing, it is then normalized. During normalization, the data is rescaled to fall between $-1$ and 1. The data is normalized using the min-max scaler function given in Equation (9).

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$  \hspace{1cm} (9)

where $x$ is the original value and $x'$ is the normalized value.

3.2.2. Hyper-Parameter Testing and Fine Tuning

During the fine-tuning and hyper-parameter testing, the model was trained for various epochs and the training and validation losses were calculated. Subsequently, the training and validation loss was plotted to visually analyze the model’s performance. Figure 7 depicts the training and validation loss for 50, 100, and 150 Epochs, respectively.

![Figure 7. Training and Validation loss.](image)

For 50 epochs, the graph depicts that the model is underfit since the validation and training plots do not converge. For 150 epochs, the model was getting overfit and could not be effectively generalized. The best-fit model was 100 epochs and a batch size of 50, whereas the sequence size had been set to 10.

The optimizer used for the model was ADAM. The reason for selecting Adam as an optimizer is that it has a learning rate of 0.001 by default. That is, it is neither too small nor too high and works best in the case of LSTM. A stacked LSTM model detailed in Table 2 was then defined.
Table 2. Characteristics of the LSTM model.

<table>
<thead>
<tr>
<th>Type</th>
<th>Stacked LSTM</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. of Hidden Layers</td>
<td>Two (02)</td>
</tr>
<tr>
<td>No. of Memory Units</td>
<td>Layer 01: 128</td>
</tr>
<tr>
<td></td>
<td>Layer 02: 64</td>
</tr>
<tr>
<td>Optimizer</td>
<td>Adam</td>
</tr>
<tr>
<td>Batch Size</td>
<td>50</td>
</tr>
<tr>
<td>No. of Epochs</td>
<td>100</td>
</tr>
</tbody>
</table>

To make the model work effectively in real scenarios, all analyses were conducted on the actual time series data and in the time domain. Additionally, the cleansed data from the sensors were directly input into the model, contrary to other studies where either time-domain features or frequency-domain features were used as input to the model [37–41]. These statistical analyses and statistical feature extractions cannot be effectively carried out on partial data, especially in cases where prognostics are performed in real-time, and limited SCADA data is available at any time. Furthermore, LSTM has a competitive advantage where the need for feature engineering and extensive domain knowledge is minimized [43]. Hence, instead of extracting features, raw data from the sensors were directly input into the LSTM model.

3.3. Evaluation Metrics

The performance of the proposed methodology was evaluated using Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Normalized Mean Absolute Error (NMAE), and Mean Absolute Percentage Error (MAPE) metrics given by Equations (10)–(13), respectively.

\[
RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2} 
\]

(10)

\[
MAE = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|
\]

(11)

\[
NMAE = \frac{1}{n \sigma^2} \sum_{i=1}^{n} |y_i - \hat{y}_i|^2
\]

(12)

\[
MAPE = \frac{1}{n} \sum_{i=1}^{n} \left| \frac{y_i - \hat{y}_i}{y_i} \right|^2
\]

(13)

\(y_i\) denotes the actual values and \(\hat{y}_i\) denotes the predicted values. During the prognostics of bearing vibrational data, large errors can often result in undesirable outcomes. Because RMSE squares the difference between the actual and predicted values, it gives greater weight to the larger error values and hence is a very useful metric for the performance evaluation of the proposed model. Therefore, the comparison of the model with other similar research is carried out in terms of RMSE. However, to further verify and evaluate the performance of the model, MAE, NMAE, and MAPE have also been used as evaluation metrics.

3.4. Results and Discussion

The performance of the developed LSTM model against the proposed approach was first analyzed on Dataset 02. Figure 8a shows the plot of the predicted and actual bearing vibration values. The blue color represents the actual bearing vibrations, whereas the orange color represents the predicted values. The proposed model precisely predicted the normal values and followed the degradation trend until a fault, which was also efficiently predicted. The RMSE value recorded for the Dataset 02 was 0.0145. Considering the mean vibration value, this RMSE reflects that the error in the predicted and actual values is less.
Figure 8. Plot of bearing vibrational signals—actual (blue) vs. predicted (orange) values.

The model’s performance was further verified by testing it against Dataset 03. Figure 8b depicts the predicted and actual values of the bearing vibrations. The model performed even better and achieved a remarkably low RMSE value of 0.0102. During the testing on Dataset 03, the model was also evaluated using MAE, NMAE, and MAPE. The error values in Table 3 reflect that the model performed extremely well while predicting the bearing vibration values.

Table 3. Model evaluation using various metrics.

<table>
<thead>
<tr>
<th>Model</th>
<th>RMSE</th>
<th>MAE</th>
<th>NMAE</th>
<th>MAPE</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSTM using raw bearing vibration values</td>
<td>0.0102</td>
<td>0.0108</td>
<td>0.0002</td>
<td>0.0107</td>
</tr>
</tbody>
</table>

Removing noise in the frequency domain requires in-depth domain knowledge and can lead to the removal of significant data points that may prove useful for the model while analyzing patterns. Additionally, with LSTM, the requirement for extensive feature engineering is minimized, whereas when statistical time domain features are input into a model, they can add a biases factor, thereby affecting the prediction accuracy and limiting the generalization of the model. These issues have been efficiently addressed in this research.

The results achieved by the proposed model were then compared with similar research conducted on the IMS bearing dataset. Table 4 compares the RMSE of our research with the RMSE achieved by various models in other research that were trained and tested on the same dataset.
### Table 4. Comparison of RMSE with similar research.

<table>
<thead>
<tr>
<th>References</th>
<th>Year</th>
<th>Model</th>
<th>RMSE Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Habbouche, H. et al. [34]</td>
<td>2021</td>
<td>LSTM, Bi-LSTM, CNN</td>
<td>0.015, 0.010</td>
</tr>
<tr>
<td>Lee, K. et al. [35]</td>
<td>2018</td>
<td>Bi-LSTM, Uni-LSTM</td>
<td>0.973</td>
</tr>
<tr>
<td>Berghout, T. et al. [36]</td>
<td>2021</td>
<td>LSTM</td>
<td>0.214</td>
</tr>
<tr>
<td>Akpudo et al. [37]</td>
<td>2020</td>
<td>Gaussian Process, LSTM</td>
<td>0.015, 0.013</td>
</tr>
<tr>
<td>Yang et al. [40]</td>
<td>2020</td>
<td>LSTM, DLSTM</td>
<td>0.030, 0.010</td>
</tr>
<tr>
<td>Ding, H. et al. [46]</td>
<td>2020</td>
<td>LSTM</td>
<td>0.045</td>
</tr>
<tr>
<td>He, M. et al. [47]</td>
<td>2020</td>
<td>LSTM</td>
<td>0.025</td>
</tr>
<tr>
<td>Huang, G. et al. [48]</td>
<td>2020</td>
<td>MLSTM</td>
<td>0.766</td>
</tr>
<tr>
<td>Ge, Y. et al. [49]</td>
<td>2019</td>
<td>LSTM</td>
<td>0.020</td>
</tr>
<tr>
<td>Chen, Z. et al. [50]</td>
<td>2018</td>
<td>LSTM</td>
<td>0.109</td>
</tr>
<tr>
<td>Tang, G. et al. [51]</td>
<td>2018</td>
<td>LSTM</td>
<td>0.055</td>
</tr>
<tr>
<td>This Research</td>
<td>2023</td>
<td>LSTM</td>
<td>0.014 &amp; 0.010</td>
</tr>
</tbody>
</table>

The researchers who achieved close RMSE values, such as [34], first extracted the time-frequency features from the data and then used those features as input to predict the future state of the bearings. However, in this research, for predicting the future vibration values, normalized raw vibration values of the bearings were used as input to the LSTM model. This not only helped reduce the need for feature engineering but also resulted in a better generalization of the model.

Table 5 compares the prediction accuracy of other machine learning and deep learning models with this research.

### Table 5. Comparison of prediction accuracy with other machine learning models.

<table>
<thead>
<tr>
<th>References</th>
<th>Classifier</th>
<th>Testing Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>[52]</td>
<td>KNN</td>
<td>91.23%</td>
</tr>
<tr>
<td>[53]</td>
<td>SVM</td>
<td>62.5%</td>
</tr>
<tr>
<td>[54]</td>
<td>DNN with temporal coherence</td>
<td>94.9%</td>
</tr>
<tr>
<td>[55]</td>
<td>Compact 1D CNN</td>
<td>97.13%</td>
</tr>
<tr>
<td>This Research</td>
<td>LSTM using raw bearing vibration values</td>
<td>98.93%</td>
</tr>
</tbody>
</table>

The results reflect that our proposed methodology and model achieved better results in terms of RMSE than other research conducted on the same dataset. Also, the model developed in this research achieved better prediction accuracy than other deep learning and machine learning models when tested against the same dataset.

#### 3.5. Generalization Capability of the Model

To evaluate the generalization capability of the model, it was tested against real vibration data generated by other electro-mechanical systems, including hydro and wind power turbines. The first set of bearing vibration data was acquired from the SCADA system installed at Neelum-Jehlum Hydro Power Project (NJHPP) Pakistan, having a total generation capacity of 969 MW. The second data set, publicly available [56], was acquired from a wind power project (WPP) operating in northern Sweden.

The plots depicted in Figure 9a,b again show that the model not only effectively predicted the bearing vibration values but also followed the trend in the data. Furthermore, a low RMSE value of 0.11 and 0.12 was recorded for the NJHPP and WPP datasets, respectively. The test results show that irrespective of the data source, the model can effectively predict the bearing vibration values.


**References**


33. Khan, A.; Hwang, H.; Kim, H.S. Synthetic Data Augmentation and Deep Learning for the Fault Diagnosis of Rotating Machines. Mathematics 2021, 9, 2336. [CrossRef]


41. Wu, H.; Huang, A.; Sutherland, J.W. Avoiding Environmental Consequences of Equipment Failure via an LSTM-Based Model for Predictive Maintenance. Procedia Manuf. 2020, 43, 666–673. [CrossRef]

42. Shershinsky, A. Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. Phys. D Nonlinear Phenom. 2020, 404, 132306. [CrossRef]


Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.