Power System Technology
ISSN:1000-3673

Received: 16-09-2024 Revised: 05-10-2024 Accepted: 22-11-2024

Early Fault Detection in Hydroelectric Power Plants Using Bi-LSTM Autoencoders and Predictive Maintenance Frameworks

Raymond Tachago*[1], Tekapso Yann*[1], Desiray Sua[2], Reine Dontsa[1], Roger Tchuidjan[1]

1 Department of Electrical and Telecommunications Engineering, National Advanced School of Engineering Yaounde, University of Yaoundé I, Yaoundé, Cameroon.

2 HydroMekin Development corporation, Yaoundé, Cameroon

Abstract: - This paper proposes a predictive maintenance framework for power plants leveraging Bidirectional Long Short-Term Memory (Bi-LSTM) Autoencoders validated on data from a real-world power plant. The framework preprocesses sensor data through interpolation, and MinMax scaling. By learning temporal dependencies, the Bi-LSTM Autoencoder detects anomalies through reconstruction errors, enabling robust health monitoring. Health indices are computed using probabilistic methods and enhanced with thresholds and smoothing techniques for real-time reliability. Validation across datasets representing ideal, normal, and faulty conditions demonstrates the framework's effectiveness in early anomaly detection and system health evaluation. Notably, this framework enables tracking the plant's state and identifying faults up to 34 hours before their occurrence, with an accuracy of 96.8% showcasing the framework's potential to reduce unplanned outages, optimize costs, and enhance system reliability, making it invaluable for proactive monitoring in industrial applications.

Keywords: Anomaly Detection, Predictive Maintenance, Bidirectional LSTM Autoencoder, Health Monitoring, Real-Time Monitoring, Power plant, Machine Learning, Operational Efficiency.

1. Introduction

Industrial systems, particularly in power generation, are highly complex and operate under stringent reliability and safety requirements. Unplanned outages in these systems can lead to significant financial losses, reduced operational efficiency, and safety hazards. To address these challenges, predictive maintenance strategies have gained prominence as they allow early detection of anomalies, enabling proactive interventions that reduce downtime and maintenance costs. While traditional approaches, such as rule-based diagnostics and threshold alarms, have been used for anomaly detection, their inability to handle the dynamic, temporal, and nonlinear patterns of modern industrial systems limits their effectiveness [1] [2].

Recent advancements in machine learning, particularly deep learning, have introduced sophisticated models for anomaly detection and system health monitoring. LSTM networks and their variants, such as Bi-LSTM networks, are widely recognized for their ability to capture sequential dependencies in time-series data. LSTM Autoencoders, in particular, have been

Power System Technology
ISSN:1000-3673
Received: 16-09-2024 Revised: 05-10-2024 Accepted: 22-11-2024

extensively applied to detect deviations from normal operational behavior through reconstruction errors, providing a robust framework for anomaly detection [3] [4]. Despite their widespread use, the majority of these studies are limited to simulations or benchmark datasets, leaving a gap in real-world industrial applications.

This paper presents an application of a Bi-LSTM Autoencoder-based predictive maintenance framework, validated on data from a real-world power plant. Unlike many existing studies that rely on synthetic or controlled datasets, our work demonstrates the feasibility and effectiveness of advanced anomaly detection methods in operational industrial settings.

What distinguishes this study is its application to actual operational data, representing conditions such as normal operation, and major failures. Validation results show that the framework is capable of detecting faults up to 34 hours before their occurrence, providing actionable insights that significantly improve reliability and reduce unplanned outages in power plants. This real-world validation bridges the gap between theoretical developments and practical deployment, offering a scalable, efficient, and robust solution for predictive maintenance in modern industrial systems. The rest of the paper unfolds as follows. Section II discusses the powerplant dataset. Section III elaborates on the main components of our model. Section IV describes the model architecture explaining the anomaly detection mechanism. The proposed model is implemented on the hydroelectric powerplant dataset in section V and the results are discussed in section VI. Finally, the paper is concluded in section VII.

2. Dataset

Our dataset is obtained from a hydroelectric powerplant, which includes 18759 data records collected with a one-hour interval and covering different states of the plant, 8759 data records were collected when the plant was considered to be working under ideal conditions for the year 2014 and 10,000 data records were collected through out the plant life covering the plant under different working conditions. This is a multivariate time-series dataset with one environmental variable which is the level of water in the damp and 19 physical variables collected from plant sensors which are listed as follows: temparetures of Reverse bearing pad 1 and 2; Air supply pressure to generator; Field voltage; field current; unit speed; active and reactive power; temparetures of air cooler 1 and 2; flow rate; oil level; back oil tank level; water gate degree of opening 1 and 2, generator oil tank pressure; generator oil tank level, shroud pressure . four of these parameters are illustrated for the first 15 days in figure 1.

Power System Technology
ISSN:1000-3673
Received: 16-09-2024 Revised: 05-10-2024 Accepted: 22-11-2024

Figure 1:Visualization of the four features of the hydroelectric power plant

From a preliminary analysis of the available data, we identified several constraints to take into account: (i) the lack of a proper record of failures for each part of the plant; (ii) the presence of only time series, with different levels of autocorrelation among the variables (iii) the presence of variables with little or no variations in their dataset. The first point prevented us from using supervised techniques; the second one posed some troubles to other popular approaches available in literature, such Ordinary Least Squares Regression [5], Principal Component Analysis [6], Clustering Algorithms [7], Autoregressive Integrated Moving Average Models [8], Granger Causality Tests [9]; and the third point permitted us to reduce physical variables moving from 19 to 11.

3. Bi-LSTM Autoencoder

A Bi-LSTM Autoencoder model is designed for early anomaly detection by leveraging a neural network architecture that employs Bi-LSTM units to encode and decode input timeseries data. The primary objective of the model is to learn a compact representation of the input data during the encoding phase and subsequently reconstruct the original timeseries data in the decoding phase. The architecture consists of two key components: the encoder and the decoder. The encoder processes the input timeseries data and transforms it into a lower-dimensional representation using Bi-LSTM layers, effectively capturing the temporal dependencies in the data. The decoder then takes this encoded representation and reconstructs the original timeseries data through another set of Bi-LSTM layers. Anomalies are detected by calculating the reconstruction error, which is the difference between the original input data and the output produced by the decoder. Data points with high reconstruction errors are flagged as anomalies because the model, trained on normal data patterns, identifies deviations from these learned patterns as unusual behavior. The Bi-LSTM autoencoder model is trained on a dataset consisting of data collected during ideal plant functioning only. During inference, the model

Power System Technology
ISSN:1000-3673
Received: 16-09-2024 Revised: 05-10-2024 Accepted: 22-11-2024

reconstructs new timeseries data and identifies any anomalies based on the reconstruction error. LSTM-based architectures are well known for its superior ability to deal with the kind of dataset we are working with, as well as its ability solve the notorious vanishing point problem [10] . Also, this architecture has been extensively used in anomaly detection of timeseries data such as Stacked-LSTM [11], Stacked LSTM-SVM [12], Contractive LSTM [13], Variational LSTM AE [14], Observer-based LSTM AE [15] and CNN-based LSTM [16]. By using a Bi-LSTM Autoencoder based approach we can enhance the accuracy of predictions by training the network not only on past data to forecast the future but also on future data to forecast the past [17]. The components of a Bi-LSTM Autoencoder are explained in the following subsections.

3.1. LSTM

An LSTM cell is the building block of our proposed model. LSTM is considered an upgraded form of RNN. One of the main limitations of RNNs is the vanishing gradient problem, which occurs when the gradients of the network weights become too small during backpropagation, causing the network to stop learning. LSTM networks solve this problem by introducing gates that regulate the flow of information in the network, allowing it to selectively remember or forget information over long periods. Another advantage of LSTM networks is their ability to capture longterm dependencies in sequential data. Unlike RNNs, which may forget important information over long sequences, LSTMs can remember important information for much longer periods, making them more effective at capturing complex patterns and relationships in sequential data. Figure 2 shows the components of a typical LSTM unit, which consists of a cell, an input gate, an output gate, and a forget gate. The three gates control the flow of information into and out of the cell, and the cell remembers values across arbitrary time intervals. The detailed working mechanism of an LSTM unit can be explained through the elaboration of its three gates: forget gate, input gate and output gate [18].

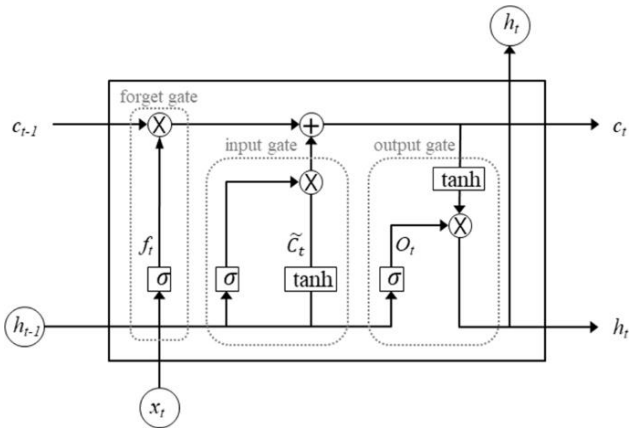

Figure 2: An LSTM cell with three gates

Power System Technology
ISSN:1000-3673
Received: 16-09-2024 Revised: 05-10-2024 Accepted: 22-11-2024

Forget gate: The forget gate determines which information from the previous cell state (c_{t-1}) should be discarded. It uses a sigmoid activation function to produce values between 0 (completely forget) and 1 (completely keep) as illustratd in equation 1.

$$f_t = \alpha(W_f.[h_{t-1}, x_t] + b_f \quad (1)$$

Where :

- f_t: Forget gate output at time t.
- α: Sigmoid activation function.
- W_f: Weight matrix for the forget gate.
- $[h_{t-1}, x_t]$: Concatenation of the previous hidden state h_{t-1} and the current input x_t.
- b_f: Bias vector for the forget gate.

The forget gate's output f_t is multiplied with the previous cell state c_{t-1} to decide what to retain or discard.

Input gate: The input gate decides which new information should be added to the cell state (c_t) from the current input and the previous hidden state.

There are two components :

1. The sigmoid function determines which values to update:

$$I_t = \alpha(W_i.[h_{t-1}, x_t] + b_i \quad (2)$$

2. The candidate values are generated using the tanh activation function:

$$\dot{c}_t = tanh(W_c.[h_{t-1}, x_t] + b_c \quad (3)$$

The new cell state is updated by combining the forget gate's output and the input gate:

$$c_t = f_t.c_{t-1} + i_c.\dot{c}_t \quad (4)$$

Where :

- I_t: Input gate output.
- c_t: Candidate cell state.
- W_c: Weight matrices for the input gate and candidate state.
- b_i, b_c: Bias vectors for the input gate and candidate state.

Output Gate: The output gate determines which part of the updated cell state should be passed to the hidden state.

Power System Technology
ISSN:1000-3673
Received: 16-09-2024 Revised: 05-10-2024 Accepted: 22-11-2024

$$o_t = \alpha(W_o.[h_{t-1}, x_t] + b_o \quad (5)$$

The hidden state is then calculated by applying the tanh activation function to the cell state and multiplying it by the output gate's output:

$$h_t = o_t \tanh(c_t) \quad (6)$$

Where :

- o_t: Output gate output.
- W_o: Weight matrix for the output gate.
- b_o : Bias vector for the output gate.

3.2. Bi-LSTM

An LSTM layer is made up of a sequence of LSTM cells, and the input data is processed only in a forward direction. On the other hand, Bi-LSTM includes an additional LSTM layer that processes the data in a backward direction, as shown in Figure 3. Training a Bi-LSTM network is equivalent to training two separate unidirectional LSTM networks. One of these networks is trained on the original input sequence, while the other is trained on a reversed copy of the input sequence. This approach provides the network with more contextual information, leading to faster and more comprehensive learning of the problem.

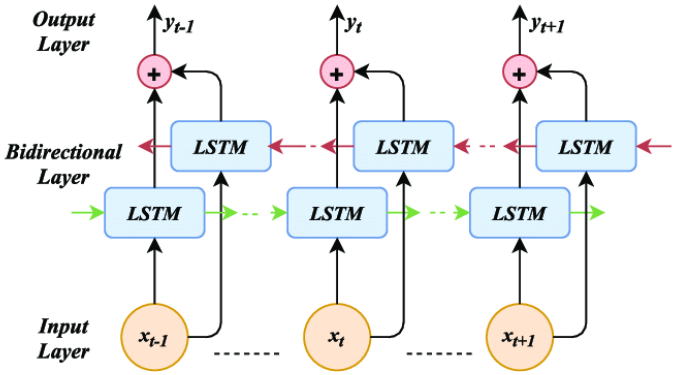

Figure 3: An unfolded view of Bi-LSTM

3.3. Autoencoder

Autoencoders are a neural network architecture utilized for unsupervised learning, data compression, dimensionality reduction, and data generation. The fundamental concept behind autoencoders involves acquiring a compressed representation of the input data (encoding) that can later be utilized to rebuild the original data (decoding) with minimum loss of information. An input layer, an output layer, and multiple hidden layers make up a conventional Autoencoder. A simple Autoencoder model is shown in Figure 4 which can be explained by

Power System Technology
ISSN:1000-3673
Received: 16-09-2024 Revised: 05-10-2024 Accepted: 22-11-2024

the encoder part, decoder part, and the reconstruction error (loss) between the input and output data [18].

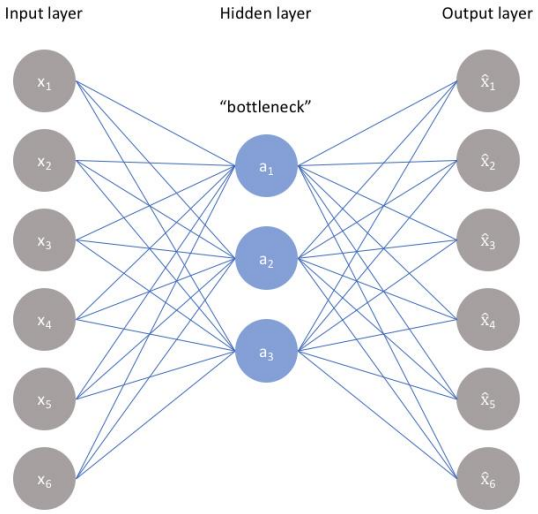

Figure 4: A basic Autoencoder architecture

Encoding: The encoding operation transforms the high-dimensional input data, represented as $x \in R^m$, into a lower-dimensional representation h in the bottleneck layer. The operation is expressed as:

$$h = f_1(W_i x + b_i) \quad (7)$$

Where:

- x: The input data vector.
- W_i: The weight matrix for the encoder.
- b_i: The bias vector for the encoder.
- f_1: The activation function
- h: The low-dimensional representation of the input data.

Decoding: The decoding operation uses the low-dimensional representation h generated by the encoder to reconstruct the original input x. The reconstructed data is represented as $\hat{x}$ and is calculated as:

$$\hat{x} = f_2(W_j x + b_j) \quad (9)$$

Where:

- $\hat{x}$: The reconstructed output data.
- W_j: The weight matrix for the decoder.

Power System Technology
ISSN:1000-3673
Received: 16-09-2024 Revised: 05-10-2024 Accepted: 22-11-2024

- b_j: The bias vector for the decoder.
- f_2: The activation function

Reconstruction Loss: The reconstruction loss L $(x, \hat{x})$ measures the difference between the original input x and the reconstructed output $\hat{x}$. This loss is minimized during training to improve the model's reconstruction capabilities. The reconstruction loss is defined as:

$$\mathrm{L}\ (x - \hat{x}) = \frac{1}{n}\sum_1^n \|\hat{x}_t - x_t\| \quad (10)$$

Where:

- L $(x, \hat{x})$: The loss function, typically Mean Squared Error (MSE).
- n: The number of data record in the dataset.
- x_t: The original input data record at time t.
- $\hat{x}_t$: The reconstructed data at time t.

4. Methodology

In this section, we present our suggested framework, which analyzes timeseries data to identify anomalies using an Autoencoder equipped with a Bi-LSTM architecture. Our proposed model architecture is illustrated in Figure 5. The architecture is composed of four stages: a timeseries input, a Bi-LSTM encoder, a Bi-LSTM decoder, and an anomaly detection module. These stages are discussed in the following:

4.1. Timeseries Data Input:

The original dataset for health evaluation in power plant conditions is first processed to a suitable format for input into the LSTM cells. The data undergoes preprocessing, including linear interpolation to handle missing values and MinMax scaling to normalize all features to a range between 0 and 1 as shown in Equation 11 aiming to handle missing values and improve model stability and convergence speed respectively.

$$x'' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (11)$$

Where ,

- X′′ is the scaled value.
- x is the original value.
- x_{min} is the minimum value in the dataset.
- x_{max} is the maximum value in the dataset.

Power System Technology
ISSN:1000-3673
Received: 16-09-2024 Revised: 05-10-2024 Accepted: 22-11-2024

After this step, the data is reshaped into 3-dimensional arrays with the structure (samples, timesteps, features) Here, "samples" refers to the total number of observations in the dataset, "timesteps" denotes the number of sequential time intervals included in each sample, and "features" represents the number of variables recorded at each timestep.

In our implementation, the training dataset consists of 6131 samples, with each sample containing 10 timesteps and 11 feature per timestep, resulting in an input shape of (6131,10,11). Similarly, the other datasets are reshaped to ensure uniformity in dimensions for compatibility with the LSTM Autoencoder. This structure enables the LSTM to process temporal sequences, capturing patterns and dependencies across the time intervals.

Each sample in the dataset is effectively a 2-dimensional array of shape (timesteps, feature). For instance, a sample Xi at any time t is represented as a sequence of inputs $[x_1, x_2, \ldots, x_t]$, where x_t corresponds to the feature value at time t. This format ensures that the model can effectively analyze sequential data to reconstruct the input and detect deviations, which are then used for evaluating the health index and identifying anomalies. This systematic approach to data preparation ensures consistency and enables the model to focus on capturing temporal patterns critical for anomaly detection and health evaluation.

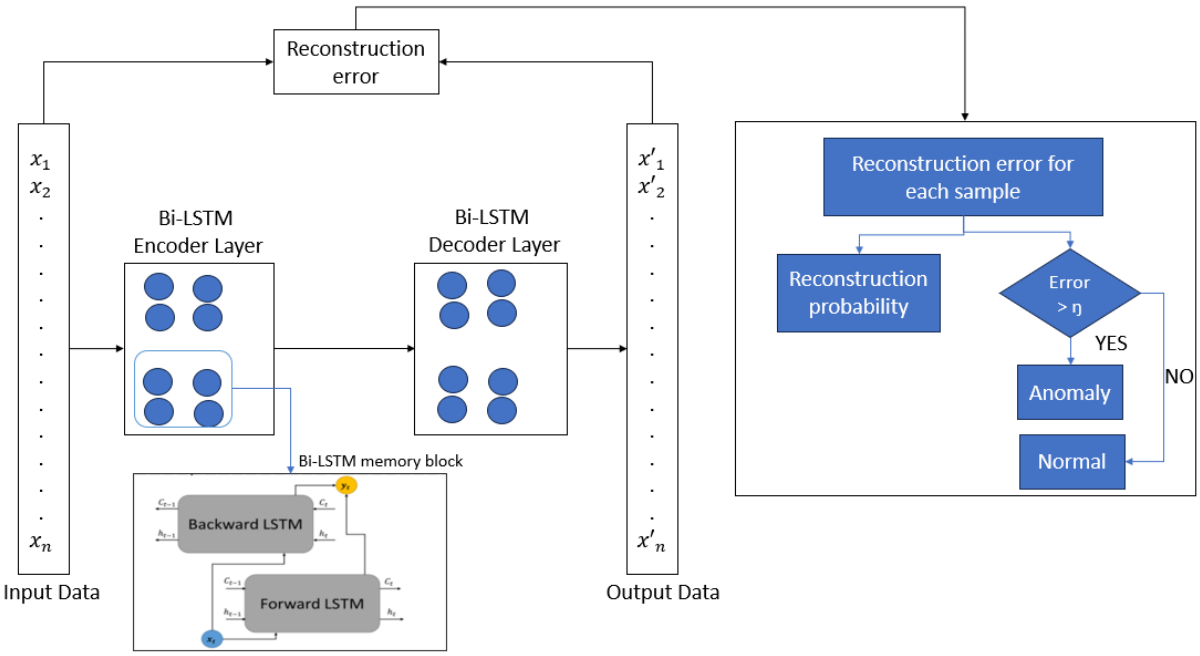

Figure 5: Proposed Bi-LSTM Autoencoder architecture for health index and anomaly detection

4.2. Bi-LSTM Encoder

The encoder in our model processes sequences of input timeseries data and converts them into a fixed-size vector. By incorporating Bi-LSTM layers, the encoder effectively captures

Power System Technology
ISSN:1000-3673
Received: 16-09-2024 Revised: 05-10-2024 Accepted: 22-11-2024

temporal dependencies in both forward and backward directions, making it highly efficient at learning patterns from the data. This bidirectional processing allows the model to leverage information from both the past and the future at each timestep, enhancing its ability to represent sequential data accurately.

The encoder architecture consists of two Bi-LSTM layers. The first Bi-LSTM layer contains 64 cells and is configured to return sequences, meaning it outputs the hidden states for each timestep in the input sequence. This layer ensures that temporal dependencies across the entire input sequence are preserved in both forward and backward directions. The second Bi-LSTM layer contains 32 cells and is configured to return only the final hidden state, reducing the sequence to a single fixed-size vector representation. This vector serves as a compressed summary of the input sequence, capturing both past and future dependencies within the data.

4.3. Bi-LSTM Decoder

The decoder LSTM generates a fixed-size input sequence from the reduced representation of the input data in the hidden layer. While reconstructing, it tries to minimize the difference between the original and reconstructed data (reconstruction error). These reconstruction errors can be utilized to establish a threshold which can in turn identify anomalies. The input to the decoder is the output from the encoder in a reduced dimension. The decoder layer is designed to unfold the encoding. Therefore, the decoder layers are stacked in the reverse order of the encoder. The first decoder layer thus has 32 LSTM cells and the second decoder layer in our model has 64 LSTM cells. The final output from the decoder is the reconstruction of the input to our model.

4.4. Anomaly Detection

An observation that deviates from the majority of the data might be referred to as an anomaly [18]. To determine how far an observation deviates, and to assess the state of our plant at any time, a health index and a threshold needs to be established as described in the following:

HI: It serves as a measure of the system's condition, providing an indication of how well the input timeseries data aligns with the expected normal patterns. The Health Index is derived from the reconstruction errors produced by the LSTM Autoencoder. When the model processes input data, it attempts to reconstruct the sequence based on the learned representations of normal behavior. The difference between the original input and the reconstructed output, quantified as the reconstruction error, forms the basis for calculating the HI.

The reconstruction error is first calculated as the mean squared error between the original input and the reconstructed output for each sample in the dataset. This error is then normalized using a probabilistic transformation to assess the likelihood of the observed error under normal operating conditions. Specifically, we calculate a reconstruction probability based on the distribution of errors observed during training such that a value closer to 0 suggest the presence

Power System Technology
ISSN:1000-3673
Received: 16-09-2024 Revised: 05-10-2024 Accepted: 22-11-2024

of anomalies or deviations from normal behaviour and a value closer to 1 suggest a healthy state as illustrated in Equation 12.

$$P(Error) = 1 - exp(-\frac{1}{2}(\frac{Error-\mu}{\sigma})^2) \quad (12)$$

Where,

- P(Error) is the reconstruction probability.
- Error is the reconstruction error for a data point.
- μ is the mean reconstruction error, computed from the training dataset.
- σ is the standard deviation of the reconstruction error, also computed from the training dataset.

Threshhold: To account for variations in reconstruction error across different datasets and operational scenarios, a threshold is incorporated into the model. The threshold calculated based on the mean and standard deviation of the reconstruction loss during training. Data points with reconstruction probabilities falling below this threshold are flagged as anomalies as illustrated in Equation 13.

$$Threshold = \mu + k \cdot \sigma \quad (13)$$

Additionally, a smoothed version of the Health Index is calculated using a simple moving average (SMA) to reduce noise and provide a clearer indication of the system's condition over time as illustrated in Equation 14.

$$SMA_t = \frac{1}{w}\sum_{i=0}^{w-1} x_{t-i} \quad (14)$$

Where :

- SMA_t: The smoothed value at time t.
- x_{t-i}: The original value at time t−i.
- w: The window size (number of data points included in the average).

This approach allows the model to effectively monitor the system's health and detect deviations that could indicate potential failures. By continuously evaluating the Health Index, our model ensures real-time anomaly detection and robust health monitoring, providing valuable insights for predictive maintenance and system optimization.

5. Implementation

In this work, the power plant dataset dedicated to training consisted of a total of 8759 samples or data points. 70% of the data is used for training and 30 % for validating the model; and over 10 000 samples collected from the power plant is used for testing. Furthermore, the model has been trained with data collected when the plant was working under ideal conditions. Each

Power System Technology
ISSN:1000-3673
Received: 16-09-2024 Revised: 05-10-2024 Accepted: 22-11-2024

sample has a timestep period of 10. The shape of the training data is thus 6131×10×11. The shape of the test data is 10,000×10×11, which indicates that we have a total of 10000 samples in the test set, each sample has a timestep period of 10 with 11 features. It is important to note that, in the case of the test set, the samples contain both positively and negatively labeled response values.

The encoder part of the network has two Bi-LSTM layers; the first layer has 64 LSTM cells whereas the second layer has 32 LSTM cells. For the decoder part, the first LSTM layer has 32 units and the second one has 64 LSTM units. The model is trained with a learning rate of 0.0001 and a batch size of 128. After the training is done, the learning curve for loss calculated with the training and validation dataset is shown in Figure 6.

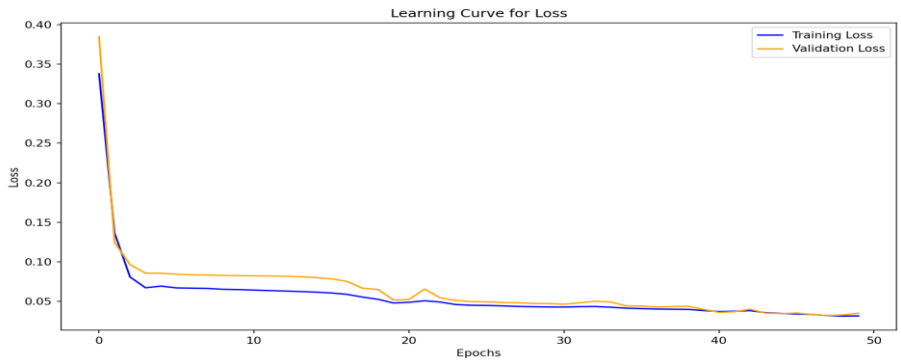

Figure 6: Learning curve for loss

The training data is also used to calculate the reconstruction loss using Equation 10. A distribution of reconstruction loss of the training set which contains no anomalies is shown in Figure 7. The threshold beyond which a datapoint can be termed as an anomaly can be set using domain knowledge and this distribution.

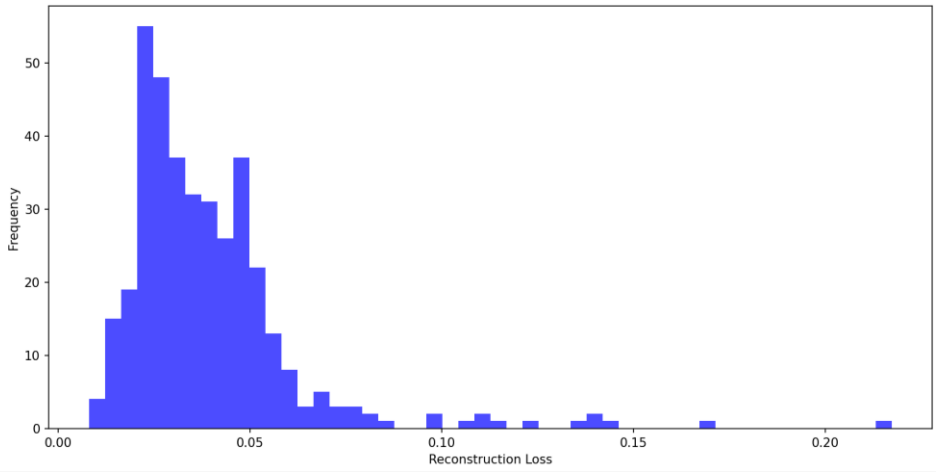

Figure 7: Distribution of reconstruction loss of the training set

Power System Technology
ISSN:1000-3673
Received: 16-09-2024 Revised: 05-10-2024 Accepted: 22-11-2024

After training the model and choosing the right threshold to detect anomalies, the test dataset is now passed through the trained model. Figure 8 illustrates the reconstruction error for our entire test dataset, we can see that the anomalous data points (orange dots) have a higher reconstruction error.

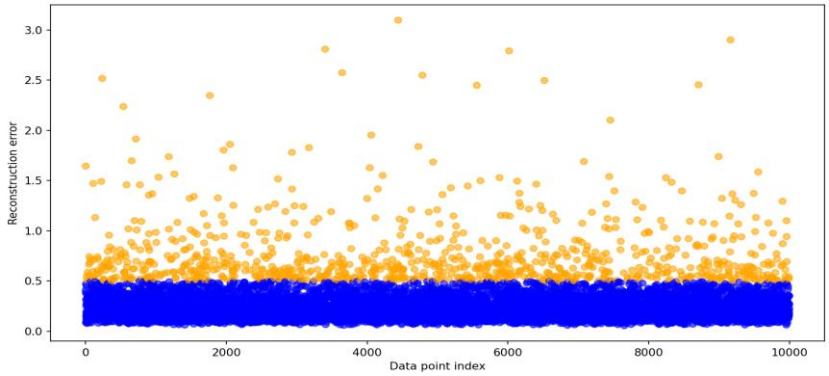

Figure 8:Reconstruction errors for test dataset

Based on company records, the reconstruction probability of three distinct state of the plant collected from the test dataset can be visualized in Figures 9,10 and 12 where we have, perfect conditions (collected after a complete preventive maintenance action in the plant), normal conditions (data collected randomly in the dataset that illustrates the plant functioning), and Faulty condition (historical data that illustrates when the plant went through a fault) respectively.

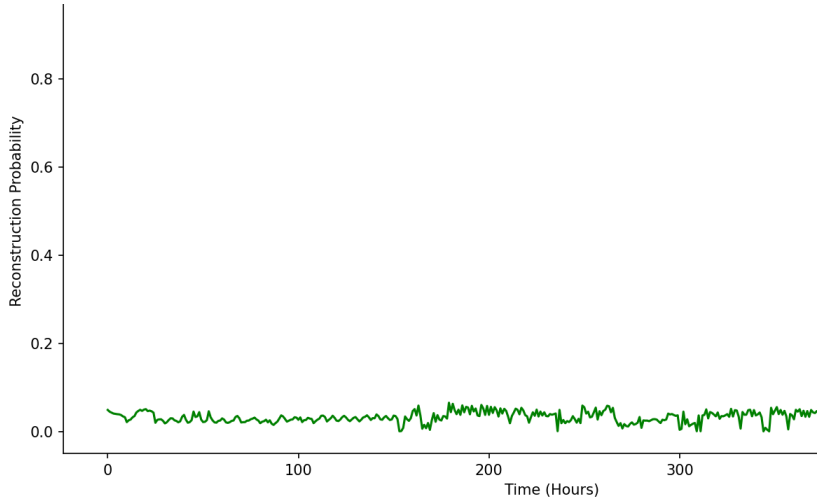

Figure 9: Health index of plant under near perfect working condition (400 hours displayed)

Figure 9 illustrates a near 0 reconstruction probability, which indicates that the plant is functioning under nearly ideal conditions illustrating the fact that the plant went through a complete preventive maintenance framework.

Power System Technology
ISSN:1000-3673
Received: 16-09-2024 Revised: 05-10-2024 Accepted: 22-11-2024

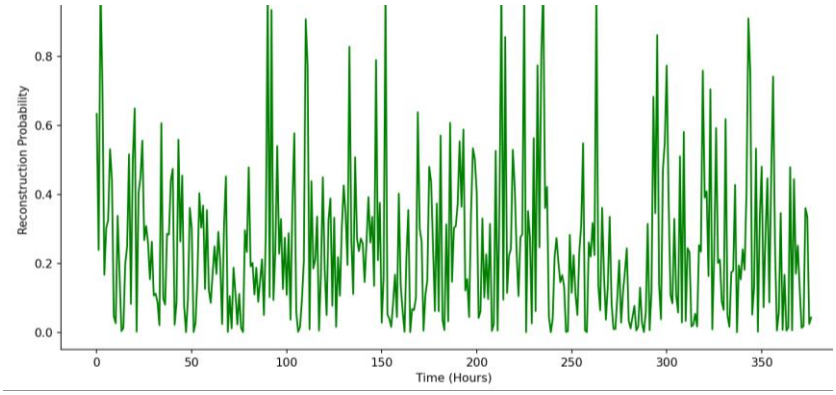

Figure 10: Health index of plant collected randomly within the test dataset with no anomaly (400 hours displayed)

Figure 10 illustrates the plant working on a random day, with no real fault clearly identified, we can observe that the curve fluctuates significantly between values close to 0 and values up to around 0.9. spikes that reach 0.9 are considered as noise, as they are mostly isolated and do not reflect the actual state of the plant. Figure 11 shows a SMA of this plot with window 50, better illustrating the plant health state during this period with a recontruction probability ranging from 0.15 to 0.35 .

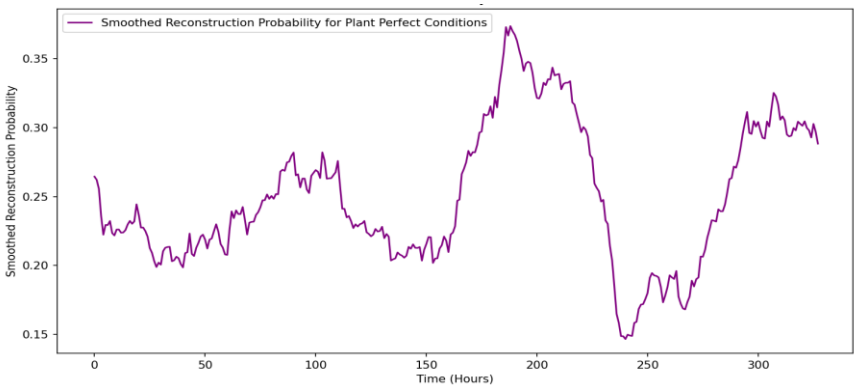

Figure 11: Smoothed health index of plant collected randomly within the test dataset with no anomaly

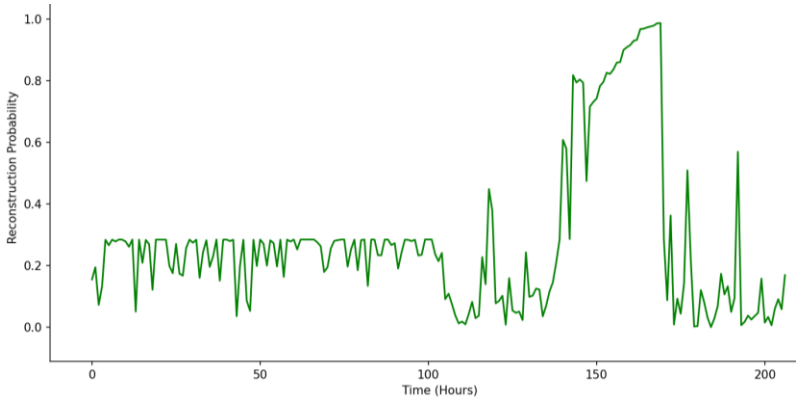

Figure 12: Health index of plant with a real fault detected at 177 hours (over 200 hours displayed)

Power System Technology
ISSN:1000-3673
Received: 16-09-2024 Revised: 05-10-2024 Accepted: 22-11-2024

Figure 12 illustrates the plant after a major failure occurred, we observe from our data set that from hour 143, the reconstruction probability rises consistently to hour 177 when the error is detected by our plant and repair measures were quickly undertaken leading the reconstruction probability to move down rapidly to 0.12 at the 178 hour.

6. Results

We used some commonly used performance metrics to evaluate the performance of our proposed Bi-LSTM Autoencoder model on the wind power dataset. These metrics are accuracy, precision, recall, and F1 score. They are expressed using the following equations:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (15)$$

$$Precision = \frac{TP}{TP+FP} \quad (16)$$

$$Recall = \frac{TP}{TP+FN} \quad (17)$$

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (18)$$

Here, TP (True Positive) refers to the count of accurately identified anomalies, TN (True Negative) refers to the count of correctly identified normal events, FP (False Positive) indicates the count of normal events that were inaccurately diagnosed as anomalies, while FN (False Negative) denotes the count of anomalies that were incorrectly identified as normal events. A confusion matrix for the test dataset is illustrated in Figure 13.

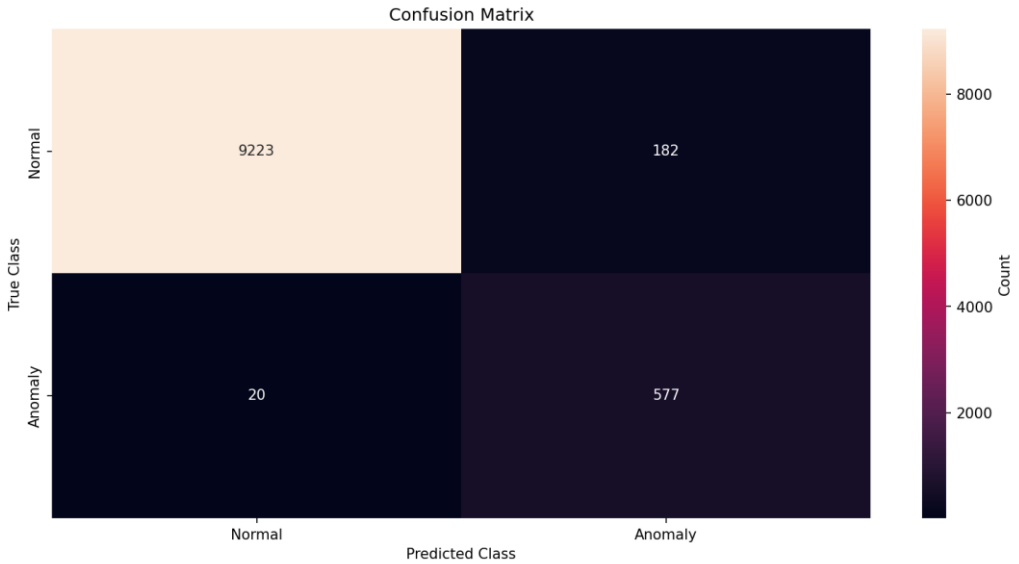

Figure 13:Confusion matrix of the test data

Power System Technology
ISSN:1000-3673
Received: 16-09-2024 Revised: 05-10-2024 Accepted: 22-11-2024

Using equations (15) - (18), we have also calculated the performance metrics for our proposed model comparing it with an LSTM autoencoder with similar architecture and parameters in Table 1.

Table 1:Performance of Bi-LSTM Autoencoder

Accuracy	Precision	Recall	F1 Score
0,968	0,721	0,885	0,794
0,933	0,498	0,843	0,701

The AUC-ROC graph in Figure 14 illustrates the effectiveness of the Bi-LSTM Autoencoder model in terms of the trade-off between true positive rate and false-positive rate. It achieved an AUC-ROC score of 0.93, which demonstrates that our proposed network is efficient and correctly identifying anomalies.

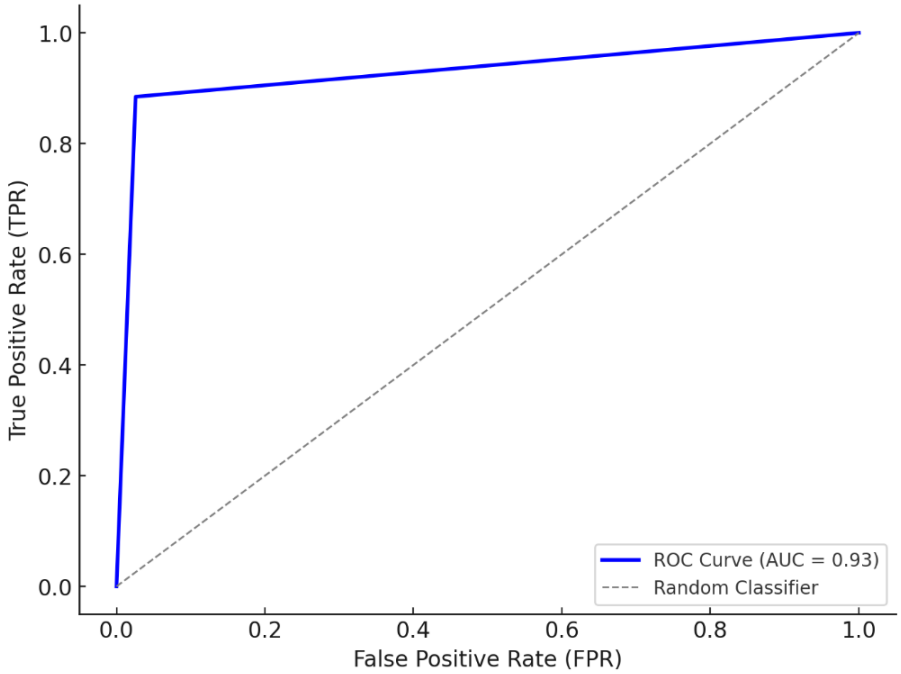

Figure 14:AUC-ROC curve of the proposed model

More over, our system has proven to be able to identify anomalies upto 34 hours before the powerplant existing maintenance framework was able to identify using the LSTM autoencoder under the same conditions we were only able to identify the fault 19 hours before it occurence, thereby demonstrating the early anomaly detection capability of the system.

Power System Technology
ISSN:1000-3673
Received: 16-09-2024 Revised: 05-10-2024 Accepted: 22-11-2024

7. Conclusion

In this paper, we have developed a predictive maintenance framework for hydroelectric power plants using Bidirectional Long Short-Term Memory (Bi-LSTM) Autoencoders. By preprocessing sensor data and leveraging the capabilities of Bi-LSTM networks, our model effectively learns normal operational patterns and detects anomalies through reconstruction errors. The health index derived from these errors, coupled with adaptive thresholding and smoothing techniques, enables real-time monitoring and early fault detection.

Our framework was validated on real-world data from a hydroelectric power plant under ideal, normal, and faulty conditions. The results demonstrate that our model can detect anomalies up to 34 hours before their occurrence, significantly earlier than traditional detection methods though this is only possible for faults that build up progressively over long hours due to the way our dataset was collected. Performance metrics indicate high accuracy (96.8%), precision (72.1%), recall (88.5%), and an AUC-ROC score of 0.93, confirming the effectiveness of our approach in practical settings. We also compared the performance with LSTM Autoencoder which was comprehensively outperformed by our proposed model

The implementation of this framework has the potential to reduce unplanned outages, optimize maintenance schedules, and enhance the overall reliability of power generation systems. By bridging the gap between theoretical developments and practical deployment, our study offers a scalable and robust solution for predictive maintenance in industrial applications.

Future work could involve integrating additional data sources, such as environmental factors or maintenance logs; collecting data over shorter intervals and exploring other deep learning architectures to further improve detection capabilities. This would enhance the model's adaptability and effectiveness across different operational scenarios, contributing to more resilient and efficient power plant operations.

Ethical Statements

This study was conducted in collaboration with a hydroelectric power plant that sought to integrate predictive maintenance into its existing maintenance framework. The data utilized in this research is proprietary and strictly confidential, ensuring the protection of sensitive operational information. Only a small anonymized sample may be made available for academic and validation purposes upon request, subject to approval by the company.

The research was conducted in full compliance with ethical guidelines and standards. The collaborating company reviewed the findings and has explicitly authorized the publication of this work. All necessary permissions and agreements were obtained to ensure the responsible use and handling of the data. The confidentiality of the plant’s operational details and sensitive information has been rigorously maintained throughout the study.

Power System Technology
ISSN:1000-3673
Received: 16-09-2024 Revised: 05-10-2024 Accepted: 22-11-2024

This ethical approach underscores the commitment of both the research team and the collaborating company to uphold data privacy, transparency, and the advancement of predictive maintenance practices in an industrial setting.

Refrences

[1] K. L. e. al, «Challenges in Industrial Predictive Maintenance,» *Industrial Engineering Journal*, vol. 42, n° %13, p. 212–220, 2023.

[2] J. S. a. L. Patel, «A Review of Machine Learning for Predictive Maintenance,» *Journal of Maintenance Science*, vol. 35, n° %11, pp. 45-62, 2022.

[3] T. B. e. al., «Autoencoders in Anomaly Detection,» *International Journal of AI*, vol. 18, n° %12, p. 101–118, 2020.

[4] A. Z. a. X. Zhang, «RNNs and Time-Series Analysis,» *Journal of Computational Models*, vol. 10, n° %12, p. 89–102, 2020.

[5] J. S. a. L. Brown, «Challenges of Multicollinearity in OLS Regression,» *Statistical Methods Journal,*, vol. 18, n° %13, pp. 45-60, 2022.

[6] M. T. a. K. White, «Limitations of PCA in High-Dimensional Data Analysis,» *Computational Statistics Journal*, vol. 34, n° %12, pp. 102-115, 2021.

[7] R. G. a. S. Wang, «Clustering Techniques and Their Applications in Data Science,» *Advances in Machine Learning*, vol. 29, n° %17, pp. 512-530, 2020.

[8] T. R. a. D. Lee, «ARIMA Models: Assumptions and Limitations,» *Time Series Analysis and Applications*, vol. 11, n° %1321-335, p. 4, 2019.

[9] K. A. a. P. Nguyen, «Limitations of Granger Causality in Nonlinear Systems,» *Journal of Time Series Econometrics*, vol. 25, n° %16, pp. 210-225, 2020.

[10] D. Pagano, «A predictive maintenance model using Long Short-Term Memory Neural Networks and Bayesian inference,» *Decision Analytics Journal*, vol. 6, n° %1100174, 2023.

[11] L. V. G. S. a. P. A. P. Malhotra, «"Long short term memory networks for anomaly detection in time series,» *in proceedings of europeen symposiumon artificial neural network, computational intelligence and machine learning* , pp. 89-94, 2009.

[12] T. Ergen and S. S. Kozat, «Unsupervised anomaly detection with LSTM neural networks,» *IEEE trans Neural Network learns,* vol. 31, n° %18, pp. 3127-3141, 2020.

[13] S. N. e. al., «Enhanced network anomaly detection based on deep neural networks,» *IEEE access*, vol. 6, p. 48231–48246, 2018.

Power System Technology
ISSN:1000-3673
Received: 16-09-2024 Revised: 05-10-2024 Accepted: 22-11-2024

[14] Y. H. a. C. K. D. Park, «A Multimodal Anomaly detector for robot assissted feeding using a LSTM based variational autoencoder,» *IEEE Robot. Autom*, 2017.

[15] N. J. a. M. W. B. Lindemann, «Anomaly detection and prediction in discrete magnufacturing based on cooperative LSTM networks,» *IEEE 16th International Conference on automation science and engineering*, pp. 1003-1010, 2020.

[16] T.-Y. K. a. S.-B. Cho, «Web traffic anomaly detection using C-LSTM and neural networks,» *Expert syt. Appl*, vol. 106, pp. 66-76, 2018.

[17] G. R. K. M. Salehi Hoshang, «Fuzzy-Biemm: An Emotion Recognition Model Based on Bidirectional Deep Learning and Extended Fuzzy Markov Model,» *Power System Technology* , vol. 48, n° %11, pp. 980-1007, 2024.

[18] J. J.-J. W. X. F. S. S. C. a. M. Y. Wei, «LSTM-Autoencoder based Anomaly Detection for,» *arXiv Prepr.*, n° %12204.06701, 2022.