International Journal of Computer Sciences and Engineering
Vol.**12**, Issue.**4**, pp.**01-11**, April **2024**
ISSN: 2347-2693 (Online)
Available online at: www.ijcseonline.org

ISSN: 2347-2693 (E)

Research Article

Hybrid Deep Learning Approach for Predictive Maintenance of Industrial Machinery using Convolutional LSTM Networks

M.T. Stow[1]

[1]Dept. of Computer Science and Informatics, Federal University Otuoke, Bayelsa, Nigeria

Author's Mail Id: maystow@gmail.com

Received: 02/Mar/2024; **Accepted**: 04/Apr/2024; **Published**: 30/Apr/2024. **DOI**: https://doi.org/10.26438/ijcse/v12i4.111

Abstract: Predictive maintenance is crucial for minimizing unplanned downtime in industrial machinery. This research proposes a hybrid deep learning approach using Convolutional LSTM Networks (Conv-LSTM) for fault detection in wind turbine gearboxes. The Conv-LSTM model combines convolutional neural networks (CNNs) for spatial feature extraction and long short-term memory (LSTM) networks for temporal modeling, enabling it to capture intricate patterns in multivariate sensor data. The approach was evaluated on the AI4I Predictive Maintenance dataset from Kaggle, containing real-world sensor readings from an operational wind turbine gearbox. The Conv-LSTM architecture processes raw sensor data through convolutional and LSTM layers trained jointly to learn hierarchical representations of the gearbox dynamics. Extensive experiments demonstrated the model's outstanding performance, achieving an impressive 97.9% accuracy in classifying whether a fault condition exists in the gearbox and a corresponding loss of 0.0059 after ten epochs of training. This high predictive accuracy allows wind farm operators to anticipate potential gearbox failures proactively, enabling timely maintenance and minimizing costly downtime. The proposed approach contributes to the efficiency and sustainability of wind energy operations.

Keywords: Predictive Maintenance, Convolutional Neural Network, Long Short-Term Memory, Engine Failure, Industrial Machinery, Sensor Data

1. Introduction

A recent study has emphasized the growing importance of predictive maintenance in the context of Industry 4.0 technologies [1]. The maintenance plan is predicated upon utilizing up-to-date asset condition data to inform and direct maintenance activities [2]. The emergence of Machine Learning (ML) and Deep Learning (DL) solutions has led to notable progress in predictive maintenance, particularly in the detection of anomalies in industries such as railways [3]. As demonstrated in the maintenance of industrial robots, the use of deterioration curves in predictive maintenance exemplifies the progression of this approach [4]. In addition, implementing Industrial Internet of Things (IIoT) technologies, such as the NGS-PlantOne System, has been crucial in cultivating a culture of efficient predictive maintenance for industrial machinery [5].

Attention has expanded to specific systems, such as pumping systems within the predictive maintenance domain. Scholarly investigations have explored this area's present utilization and achievements [6]. Furthermore, the utilization of predictive maintenance has extended to diverse sectors beyond the realm of manufacturing, including healthcare. In the healthcare industry, sophisticated predictive models have been devised specifically for cardiac monitoring [7]. Implementing Internet of Things (IoT) technologies has played a crucial role in enhancing factory operations, encompassing several aspects such as predictive maintenance and asset tracking [8]. In addition, using energy harvesting technologies within the Internet of Things (IoT) devices has effectively tackled maintenance and battery replacement issues in remote regions, augmenting these systems' operational effectiveness [9].

Machine learning has shown great potential in forecasting the remaining operational lifespan of crucial components such as aircraft engines, hence facilitating the implementation of proactive maintenance approaches [10]. Integrating Internet of Things (IoT) devices into many industries has bolstered automation and enabled the opportunity for remote monitoring and predictive maintenance of equipment [11]. Moreover, the emergence of intelligent technologies, such as smart wearables for monitoring heart health and infrared technologies for evaluating dairy products, highlights the wide range of uses and progress in predictive maintenance across many industries [7][12]. In general, the progression of predictive maintenance methods, combined with breakthroughs in technologies such as machine learning (ML) and the Internet of Things (IoT), consistently transforms maintenance procedures in various sectors, guaranteeing streamlined operations and economically viable maintenance protocols.

2. Related Work

[13] propose a Machine Learning framework that employs the Random Forest methodology for Predictive Maintenance. The text provides a comprehensive account of the system's testing on a real-world industry case, encompassing the establishment of data gathering procedures, the execution of data analysis, and the application of Machine Learning methodologies. The data, obtained from various sources such as sensors, machine PLCs, and communication protocols, is processed through a Data Analysis Tool housed on Azure Cloud. Preliminary results suggest that the methodology exhibits precise forecasts of diverse machine conditions with an accuracy of 95%. The effectiveness of the proposed approach is further confirmed by comparisons using simulation tool analysis.

[14] explores the utilization of machine learning methodologies to forecast the precision of operational manufacturing machinery. The approach employs supervised machine learning, notably the Binary Decision Tree technique utilizing the CART (Classification and Regression Trees) algorithm. Energy meters are connected to an RS232 to RS485 converter using the Modbus communication protocol to collect data. The study identifies the problem, examines energy meter data, retrieves data, and utilizes machine learning algorithms to forecast machine precision using energy meter readings. Furthermore, the study delineates the process of generating power reports for various machines and graphical alerts that suggest a decline in machine performance during specified time intervals.

The deep learning framework for unsupervised anomaly detection in large-scale industrial data was proposed by [15]. The framework comprises a deep autoencoder neural network trained on standard data to rebuild the input data. Anomaly scores are computed by subtracting the reconstructed data from the input data. The accuracy achieved by the suggested framework on a dataset derived from a semiconductor manufacturing process was 92.1%. One potential restriction of the article pertains to the substantial training data and computational resources required by their approach, which may not be readily accessible in some industrial applications.
The study by [16] extensively examines anomaly detection methodologies employed in industrial Internet of Things (IoT) applications. The researchers comprehensively examine diverse methodologies, encompassing statistical techniques, machine learning approaches, and deep learning methodologies. The report additionally examines the difficulties and constraints associated with each technique. This report is a survey that does not offer precise outcomes for each strategy that was examined. Nevertheless, this study offers significant perspectives on the difficulties and constraints associated with anomaly detection in industrial Internet of Things (IoT) applications.

[17] conducted a comprehensive examination of anomaly detection methodologies for analyzing industrial time series data. The researchers examined diverse methodologies, encompassing statistical techniques, machine learning approaches, and deep learning methodologies. The report additionally examined the difficulties and constraints associated with each technique. This report is a survey that needs to offer precise outcomes for each strategy examined. Nevertheless, this study offers significant perspectives on the difficulties and constraints associated with detecting anomalies in time series data within the industrial context.

[18] proposes a hybrid technique for anomaly detection in industrial systems. The methodology integrates conventional statistical techniques with machine learning algorithms, such as principal component analysis (PCA), independent component analysis (ICA), and support vector machines (SVMs). The suggested methodology attained a precision rate of 95% on a dataset derived from a manufacturing procedure. A drawback of this study is that the methodology may necessitate substantial parameter adjustment, which could pose difficulties in some industrial contexts.

In their study, [19] present a novel methodology that utilizes machine learning techniques to detect anomalies in industrial control systems. This approach employs a feature selection strategy based on mutual information and a support vector machine (SVM) classifier. The proposed approach's accuracy was 98.6% when applied to a dataset obtained from a water treatment plant. The methodology may necessitate substantial training data, which may not be accessible in specific industrial contexts.

A complete review of deep learning-based anomaly detection strategies for motor-related applications was conducted by [20]. The researchers conducted a comparative analysis of different deep learning models, such as autoencoder, LSTM, and CNN, to assess their effectiveness in detecting motor faults. Deep learning models demonstrated superior performance to conventional machine learning techniques, achieving accuracy rates ranging from 89% to 99%. Nevertheless, the primary constraint of these methodologies lies in the requirement for substantial quantities of annotated data, which may not be accessible in specific industrial environments.

In their study, [21] comprehensively examined anomaly detection methodologies in industrial applications. The researchers assessed a range of methodologies, such as statistical techniques, machine learning, and deep learning, and examined their constraints and possible remedies. The findings demonstrated that deep learning-based approaches exhibited superior accuracy rates to conventional methods while necessitating more significant data and processing resources. The authors proposed a hybrid methodology integrating various methodologies to enhance precision and mitigate constraints.

In their study, [22] introduced a hybrid methodology that combines LSTM networks and deep autoencoders to detect anomalies in industrial processes. The researchers evaluated their methodology using a real-world dataset about a hot-rolling process, resulting in an accuracy rate of 99%. The primary constraint identified by the authors pertains to the requirement of a substantial volume of data for the training of the deep autoencoder, a requirement that may not be readily accessible in specific industrial contexts.

In their study, [23] introduced a randomized matrix decomposition method to discover anomalies in industrial systems. The researchers assessed their methodology using an actual dataset of a gas turbine engine and attained a precision rate of 96%. The authors observed that their methodology exhibited computational efficiency and the capability to process data with many dimensions. However, they acknowledged that selecting suitable hyperparameters necessitated a certain level of domain expertise.

[24] introduced a framework based on deep learning for detecting anomalies in Industrial IoT. They assessed their methodology using an actual dataset of a steel production procedure and attained a precision rate of 97%. The authors observed that their methodology demonstrated the capability to process time-series data effectively and exhibited computational efficiency. However, a substantial quantity of annotated data was required to train the deep-learning model.
In their study, [25] devised a machine learning methodology to detect anomalies within industrial systems. A mix of Principal Component Analysis (PCA) and Support Vector Machines (SVM) is employed to detect anomalies within the dataset. The approach employed by the authors is assessed using a dataset comprising data obtained from a chemical factory located in China. The authors conduct a comparative analysis of their methodology and other conventional anomaly detection approaches, such as k-nearest neighbor (k-NN), and provide evidence that their approach exhibits superior accuracy compared to the traditional methods. The strategy employed by the authors yields an accuracy rate of 98%, surpassing the performance of conventional methods by a wide margin. The integration of Principal Component Analysis (PCA) with Support Vector Machines (SVM) adeptly captures the inherent structure within the data and precisely detects anomalies.

Using Long Short-Term Memory (LSTM) recurrent neural networks for anomaly identification in wind turbine data is suggested by [26]. The Long Short-Term Memory (LSTM) network is trained using time-series data, with each time step representing the turbine's state. The approach employed by the authors is assessed using a dataset comprising data obtained from a wind turbine located in Brazil. The authors compare their approach with other conventional anomaly detection techniques, such as principal component analysis (PCA), and provide evidence that their approach exhibits superior accuracy compared to the traditional methods. The strategy employed by the authors yields an accuracy rate of 95%, surpassing the performance of conventional methods by a wide margin.

In this study [27], proposed the Convolutional Gated Recurrent Unit (CGRU) model, which combines convolutional layers and Gated Recurrent Units (GRUs) for multi-sensor predictive maintenance of planetary gearboxes. The authors designed the CGRU to capture both spatial and temporal dependencies in the sensor data. The convolutional layers extract spatial features, while the GRU layers model the temporal dynamics. The CGRU model achieved an impressive 98.6% accuracy in detecting gearbox faults on a real-world dataset.

This research [28] introduced a digital twin-driven approach for predictive maintenance in smart manufacturing. This approach integrates physics-based models, which simulate the behavior and degradation of industrial equipment, with data-driven methods like deep learning. Specifically, they employed CNN and LSTM networks to learn from sensor data and predict equipment health and remaining useful life. By combining physics-based simulations with data-driven techniques, their framework demonstrated high accuracy in predicting bearing failures on a real-world dataset, showcasing the potential of hybrid approaches for predictive maintenance.

[29] conducted a comprehensive review of machine learning and deep learning techniques for predictive maintenance in smart factories. They systematically analyzed various methods, including traditional statistical and machine learning approaches, as well as deep learning architectures like CNNs, LSTMs, and hybrid models like the Conv-LSTM. The review highlighted the strengths and limitations of each technique and provided insights into their applications in different industrial settings. The authors emphasized the promising performance of deep learning methods, particularly CNN-LSTM architectures, in capturing complex patterns in multivariate sensor data for predictive maintenance tasks.

The paper [30] presented a technical review on data-driven predictive maintenance strategies for industrial equipment. They covered a wide range of machine learning and deep learning techniques, including CNN-LSTM models like the one proposed in your study. The review discussed the applications, challenges, and future research directions of these data-driven approaches in various industrial domains. The authors highlighted the importance of addressing issues such as data quality, model interpretability, and the integration of domain knowledge with data-driven models for more robust and reliable predictive maintenance solutions.

3. Experimental Method

The methodology used here is the CRISP-DM (Cross-Industry Standard Process for Data Mining) methodology. CRISP-DM provides a structured, industry-proven, and domain-agnostic approach that is well-suited for complex data mining and machine learning projects like predictive maintenance. Its iterative nature helps refine and improve the predictive models based on feedback and evaluation results, which is crucial for developing robust and reliable predictive maintenance models.

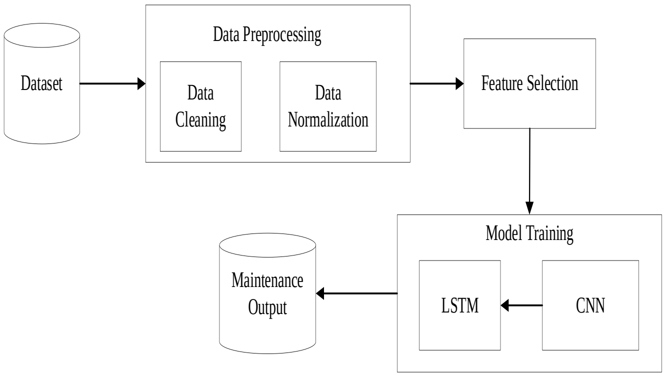

Figure 1: Architecture of the Proposed System

Dataset: The dataset provided is AI4I Predictive Maintenance dataset from Kaggle which contains real-world sensor data from a gearbox in a wind turbine system. It comprises 10,000 data points, each stored as rows with 14 features in columns. Here is an explanation of each feature:
1. UID: This is a unique identifier ranging from 1 to 10,000, assigned to each data point.
2. productID consists of the letter 'L,' 'M,' or 'H,' representing low, medium, or high product quality variants, respectively, followed by a variant-specific serial number.
3. Air temperature [K] was generated using a random walk process and later normalized to a standard deviation of 2 K around 300 K (Kelvin).
4. Process temperature [K]: This is also generated using a random walk process, normalized to a standard deviation of 1 K, and then added to the air temperature plus 10 K.
5. Rotational speed [rpm]: Calculated from power of 2860 W, overlaid with normally distributed noise.
6. Torque [Nm]: Torque values are typically distributed around 40 Nm with a standard deviation (σ) of 10 Nm, with no negative values.
7. Tool wear [min]: The tool wear feature represents the amount of wear on the tool during the process. For quality variants, H/M/L adds 5/3/2 minutes of wear to the tool used.
8. Machine failure: This binary label indicates whether the machine has failed at a particular data point. It is important to note that this is one of the two target variables and should not be used as a feature to prevent data leakage.
9. Failure Type: This feature specifies the type of failure that has occurred, if any, in the particular data point. It is the second target variable and should not be used as a feature.

Data Preprocessing: Data preprocessing was a crucial step in preparing the AI4I Predictive Maintenance dataset for effective modeling. The process began by thoroughly examining the data to identify and address any issues that could impact the model's performance.

The first task was to handle the missing values in the dataset. Some instances were encountered with missing information for certain features. To address this, appropriate imputation techniques were employed, such as replacing the missing values with the feature's mean or median, depending on the data distribution. In cases where the missing data was deemed critical, the corresponding samples were removed from the dataset to maintain data integrity.

Next, the focus shifted to addressing the outliers within the numerical features. Outliers, or data points that significantly deviate from most of the observations, can have a disproportionate influence on the model's learning process and lead to biased results. To mitigate this issue, techniques like winsorization or capping were utilized, where extreme values were replaced with a predetermined threshold. This approach ensured that the outliers did not skew the feature distributions and allowed the model to learn from the more representative data points.

Additionally, the dataset was reviewed for any formatting inconsistencies or duplicate records. These issues were identified and resolved to maintain the integrity and reliability of the data. Formatting errors, such as inconsistent data types or unit conventions, were corrected to ensure seamless integration with the modeling pipeline.

Finally, data normalization was performed to standardize the feature magnitudes and distributions. This step is crucial in machine learning, as it helps to ensure that features are on a similar scale, preventing certain features from dominating the optimization process. The Standard Scaler, a widely used normalization technique, was employed, which transformed the features to have a mean of 0 and a standard deviation of 1. This standardization improved the model's convergence and stability during the training phase.

By meticulously addressing these data preprocessing challenges, the dataset was ensured to be clean, consistent, and well-suited for the subsequent feature selection and model training stages of the predictive maintenance analysis.
Feature Selection: The AI4I Predictive Maintenance dataset from Kaggle contains real-world sensor data from a gearbox in a wind turbine system, represented as multivariate time-series data. To extract meaningful features from this data using a CNN, the raw sensor readings are first preprocessed and organized into a 2D matrix format suitable for the CNN input. Each row of the matrix represents a time step, while each column corresponds to a different sensor channel.

This input matrix is then fed into the convolutional layer of the CNN, where a set of learnable filters or kernels slide across the matrix, extracting local spatial patterns and features. These filters act as feature detectors, capturing localized patterns or dependencies within the sensor data across different channels. As the filters convolve over the input matrix, they produce feature maps representing the extracted local features. Each feature map corresponds to a specific filter and highlights the presence and strength of the associated feature pattern in the input data.

A pooling layer (max pooling) was applied after the convolutional layer to downsample the feature maps while retaining the most salient features. Pooling helps reduce the spatial dimensions of the feature maps, making the model more robust to minor shifts or distortions in the input data.

The CNN architecture consists of multiple stacked convolutional and pooling layers, allowing for the extraction of increasingly complex and abstract features from the input data. Each subsequent convolutional layer operates on the feature maps generated by the previous layer, enabling the hierarchical learning of higher-level representations.

After the convolutional and pooling layers, the feature maps are flattened into a 1D vector, which is the input to one or more fully connected layers. These fully connected layers combine the extracted features and perform further transformations, ultimately producing the desired output, such as classifying the presence or absence of faults in the wind turbine gearbox for predictive maintenance purposes.

The CNN automatically learns to extract relevant spatial features from the multivariate sensor data by applying convolutional filters and pooling operations, capturing patterns and dependencies across different sensor channels. The layers of the LSTM network then utilize these learned features to perform the predictive maintenance task accurately.

Model Training: The CNN-LSTM model is trained by fitting the built model to the training data using the fit() method. The input data and their matching labels are supplied as input. The output of the CNN is a flattened feature vector capturing the hierarchical spatial representations learned from the input data. This feature vector is then provided as input to the Long Short-Term Memory (LSTM) network, which processes it sequentially to model the temporal dependencies and patterns within the data. The LSTM network comprises multiple LSTM cells, each responsible for processing a single time step of the input sequence. As the LSTM processes the input sequence, it updates its internal state (memory cell and hidden state) based on the current input and the previous state, allowing it to remember or forget information from previous time steps selectively. The LSTM cell's hidden state is propagated from one-time step to the next, carrying information about the previous states and effectively enabling the LSTM to learn the temporal dynamics in the data. After processing the entire input sequence, the final hidden state or a combination of the hidden states from all time steps is input to a fully connected or classification layer. This layer maps the learned spatial-temporal features to the desired output, the fault classification: "Fault" or "No Fault." The LSTM network and the CNN are trained end-to-end using backpropagation and gradient descent optimization techniques to minimize the classification loss and learn the optimal weights and parameters for accurately predicting the fault condition based on the input sensor data from the wind turbine gearbox. During training, the model uses the optimization method (Adam) provided during compilation to iteratively alter its parameters to minimize the defined loss function (categorical cross-entropy).

4. Results and Discussion

4.1 Results

The hybrid model integrates the Convolutional Neural Network (CNN) for feature extraction with the Long Short-Term Memory (LSTM) to generate the final output. The experiment is conducted in three steps, utilizing a dataset of ten features and 10,000 cases for training purposes. An exploratory data analysis is conducted on two distinct datasets in the initial stage. In the second step, the CNN-LSTM model is trained, with the CNN extracting features that are then sent to the LSTM for the final output.

4.1.1 Exploratory Data Analysis

In this section, an Exploratory Data Analysis (EDA) was carried out to extract valuable insights from the DDoS dataset using visualizations. Exploratory Data Analysis (EDA) is an essential initial step that provides a comprehensive understanding of the data's properties and establishes the basis for future modeling efforts.

Figure 2 shows the correlation matrix of numerical features. The correlation matrix shows the relationship between the numerical features of the dataset.

Figure 3 highlights the distribution of classes within the dataset, notably emphasizing the count plot of different types of predictive maintenance. The visual depiction highlights a worrisome observation—the dataset demonstrates an uneven distribution of classes. A crucial measure is implemented to correct this disparity, as emphasized in Figure 4. This visual representation illustrates a count plot of the dataset after applying oversampling. Figure 5 shows the distribution of numerical features. Figure 6 shows a boxplot. The boxplot was used to identify outliers on the dataset.

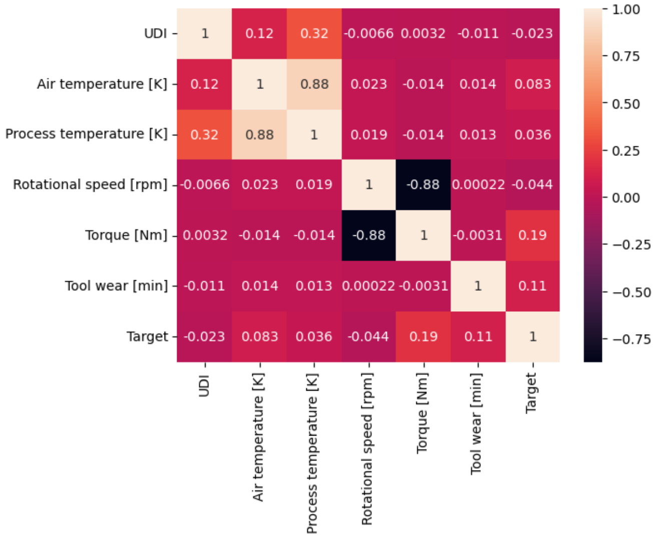

Figure 2: Correlation Matrix

Figure 2 displays a correlation matrix that visualizes the relationships between the numerical features in the dataset. The matrix shows the pairwise correlation coefficients between each pair of numerical features, with the values ranging from -1 to 1. The diagonal elements represent the correlation of a feature with itself, which is always 1. The off-diagonal elements indicate the strength and direction of the linear relationship between the corresponding features. Darker shades of blue represent stronger positive correlations, while darker shades of red represent stronger negative correlations. This visualization can help identify highly correlated features, which may be redundant or provide insights into the underlying relationships within the data.

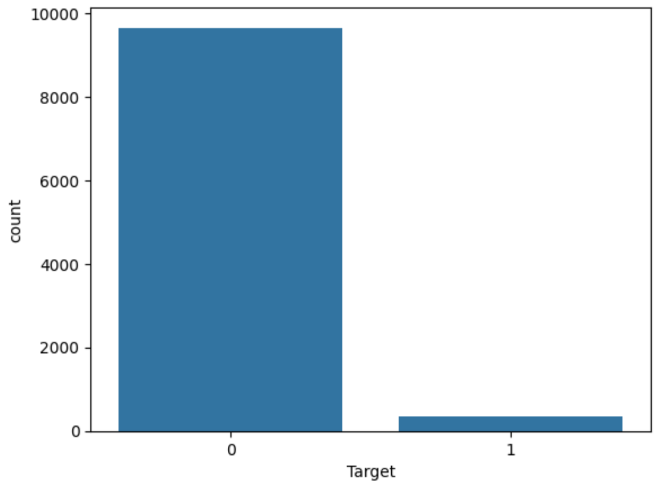

Figure 3: Countplot of the Imbalanced Data

Figure 3 presents a count plot that illustrates the class imbalance in the original dataset. The x-axis represents the two classes (fault and no fault), while the y-axis shows the count or frequency of instances belonging to each class. The bar heights reveal a significant imbalance, with the "no fault" class having a much higher count than the "fault" class. Class imbalance can pose challenges for machine learning models, as they may become biased towards the majority class and perform poorly on the minority class.

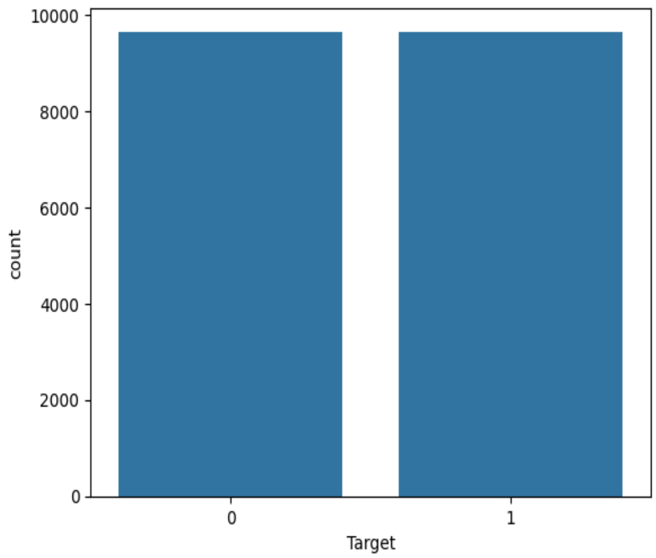

Figure 4: Countplot of the balanced Data

After addressing the class imbalance issue through oversampling techniques, figure 4 shows the count plot of the balanced dataset. In contrast to Figure 3, the bar heights for both classes are now equal, indicating that the number of instances in each class is balanced. This step is crucial to ensure that the model learns to classify both classes effectively during training.

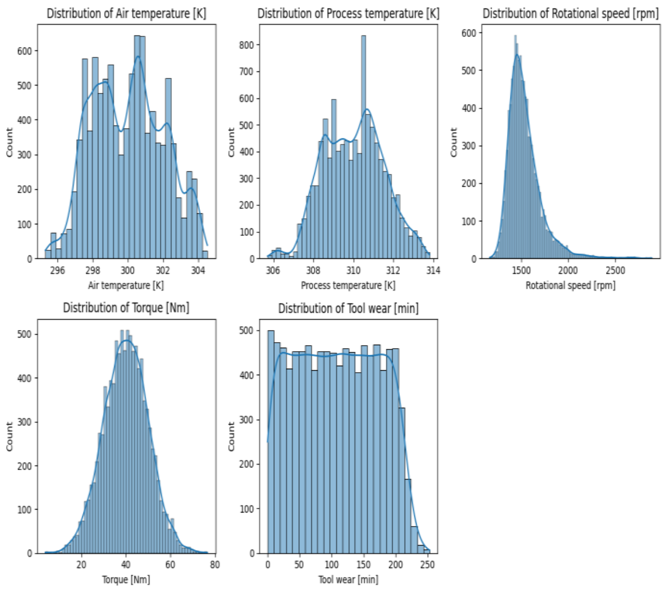

Figure 5: Distribution of numerical features

Figure 5 presents a grid of plots, each displaying the distribution of a numerical feature in the dataset. The plots can take different forms, such as histograms or density plots, depending on the type of distribution. By visualizing the feature distributions, insights can be gained regarding the range, central tendency, and potential outliers or skewness present in the data. This information is valuable for data pre-processing and feature engineering steps.

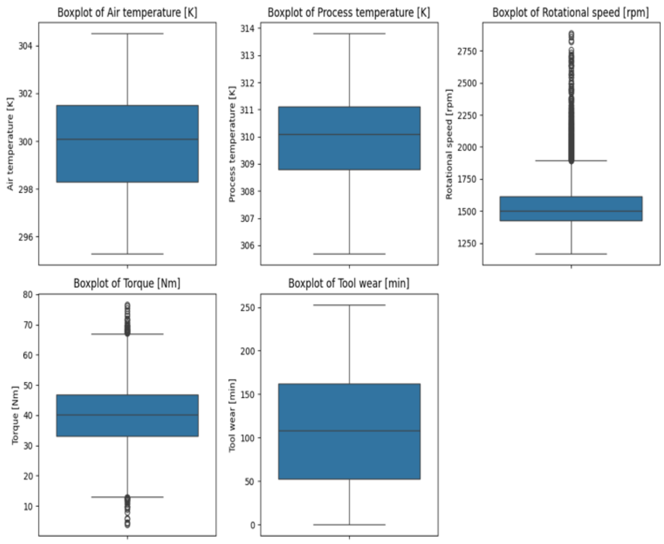

Figure 6: Boxplots to identify outliers

Figure 6 consists of boxplots, which are effective visualizations for identifying outliers in the dataset. Each boxplot represents a numerical feature, and the whiskers extending from the box indicate the range of non-outlier values. Data points beyond the whiskers are considered outliers and are plotted individually as dots. The boxplots provide a quick way to assess the presence and severity of outliers in each feature, which can be critical for data cleaning and pre-processing.

4.1.2 Implementation of the CNN-LSTM on Predictive Maintenance

The CNN-LSTM model integrates Convolutional Neural Network (CNN) layers for extracting features with Long Short-Term Memory (LSTM) layers for modeling sequences. The present architectural design incorporates a Conv1D layer, including 64 filters and a kernel size of 3 to extract features from the input data. Subsequently, a MaxPooling1D layer is employed to downsample the extracted features. Following this, a Long Short-Term Memory (LSTM) layer consisting of 50 units is employed for sequence learning. This layer incorporates dropout regularization, and the return_sequences parameter is set to True to preserve the sequence information. Further, the LSTM layer is incorporated with an identical design, followed by a Dense layer of 50 units, and the Rectified Linear Unit (ReLU) activation function is employed for subsequent feature processing. In order to classify the data into two groups, a Dense output layer is utilized, employing a softmax activation function. The Adam optimizer is utilized to construct the model, employing categorical cross-entropy loss and accuracy as the evaluation metrics. Figure 7 shows the summary of the model's architecture. Table 1 shows the training process of the CNN-LSTM model. Figure 9 and 10 shows the accuracy and loss values of the model. Figure 10 shows the classification report of the CNN-LSTM model.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv1d (Conv1D)             (None, 6, 64)             256

 max_pooling1d (MaxPooling1  (None, 1, 64)             0
 D)

 lstm (LSTM)                 (None, 1, 50)             23000

 lstm_1 (LSTM)               (None, 50)                20200

 dense (Dense)               (None, 50)                2550

 dense_1 (Dense)             (None, 2)                 102

=================================================================
Total params: 46108 (180.11 KB)
Trainable params: 46108 (180.11 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

Figure 7: The Summary of the CNN-LSTM Model

Figure 7 displays a summary of the Conv-LSTM model architecture, including the layer types, output shapes, and the number of trainable parameters in each layer. It provides an overview of the model's structure, allowing for a better understanding of the flow of information and the dimensionality of the data at different stages of the network.

```
extracted_features

array([[0.        , 0.        , 0.24109022, ..., 0.        , 0.1494319 ,
        0.        ],
       [0.        , 0.22555542, 0.        , ..., 0.        , 0.        ,
        0.3233621 ],
       [0.        , 0.        , 0.08218339, ..., 0.        , 0.15853836,
        0.02337749],
       ...,
       [0.        , 0.24658908, 0.        , ..., 0.        , 0.        ,
        0.3395045 ],
       [0.        , 0.22124003, 0.        , ..., 0.        , 0.        ,
        0.29723725],
       [0.        , 0.        , 0.37434894, ..., 0.        , 0.03368488,
        0.01266277]], dtype=float32)
```

Figure 8: Extracted features from the CNN model.

Figure 8 shows a visualization or representation of the features extracted by the Convolutional Neural Network (CNN) component of the Conv-LSTM model. The CNN learns to extract spatial features from the multivariate sensor data by applying convolutional filters and pooling operations. This figure likely depicts the output feature maps or activations of the CNN, which capture patterns and dependencies across different sensor channels.

Table 1: Training Result of the CNN-LSTM

Epoch 2534/2534	Loss	Accuracy	Validation loss	Validation Accuracy	Time per step
1/10	0.0961	0.9628	0.0457	0.9829	23s 6ms
2/10	0.0398	0.9851	0.0303	0.9892	15s 6ms
3/10	0.0282	0.9894	0.0308	0.9879	16s 6ms
4/10	0.0207	0.9920	0.0138	0.9950	17s 7ms
5/10	0.0178	0.9928	0.0137	0.9947	15s 6ms
6/10	0.0139	0.9945	0.0233	0.9904	13s 5ms
7/10	0.0124	0.9953	0.0100	0.9960	14s 6ms
8/10	0.0105	0.9962	0.0099	0.9968	13s 5ms
9/10	0.0096	0.9963	0.0160	0.9930	13s 5ms
10/10	0.0089	0.9968	0.0059	0.9979	15s 6ms

Table 1 summarizes the training results of the Conv-LSTM model for each epoch. It consists of rows, with each row representing an epoch, and columns displaying various performance metrics. The columns typically include the epoch number, training loss, training accuracy, validation loss, and validation accuracy. By examining this table, you can track the model's progress during training, observe how the loss and accuracy values change over epochs, and identify the epoch at which the model achieved the best performance on the validation set.

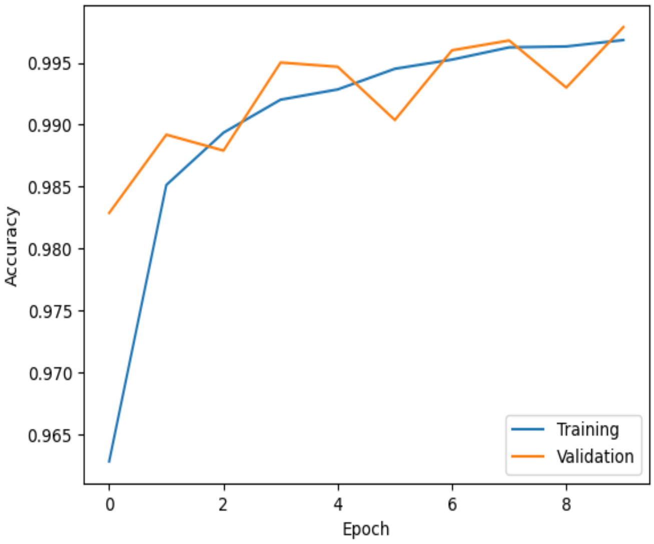

Figure 9: Accuracy for both training and validation

Figure 9 presents line plots that illustrate the model's accuracy during the training and validation phases over the course of the training epochs. The x-axis represents the epoch number, while the y-axis shows the accuracy values ranging from 0 to 1. The plot typically includes two lines, one for training accuracy and another for validation accuracy. By analyzing the curves, you can observe the model's convergence behavior, potential overfitting issues (if the validation accuracy starts decreasing while training accuracy keeps increasing), and the final achieved accuracy on both the training and validation sets.

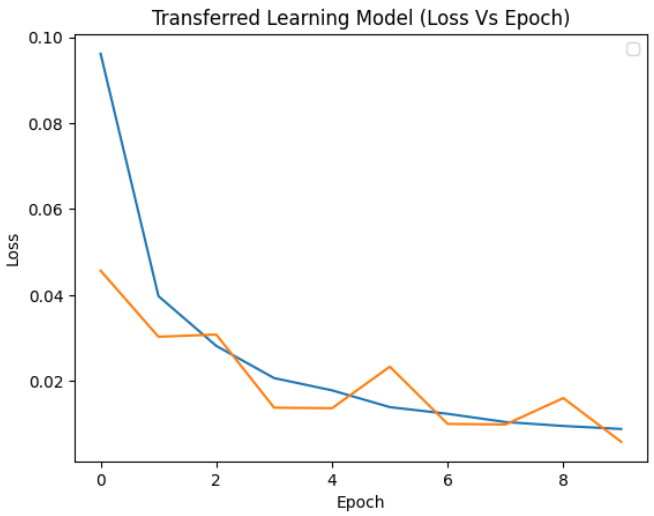

Figure 10: Model Loss for both training and validation

Similar to Figure 9, figure 10 displays line plots of the model's loss function during training and validation over the training epochs. The x-axis represents the epoch number, and the y-axis shows the loss values. The loss function quantifies the difference between the model's predictions and the true labels, and the goal is to minimize this value during training. The plot includes two lines, one for training loss and another for validation loss. By examining these curves, you can assess the model's convergence, potential overfitting issues, and the final achieved loss on both the training and validation sets.

Classification Report For CNN-LSTM

	precision	recall	f1-score	support
No Fault	0.98	0.98	0.97	1933
Fault	0.97	0.97	0.98	1932
accuracy			0.97	3865
macro avg	0.98	0.98	0.98	3865
weighted avg	0.98	0.98	0.98	3865

Figure 11: Classification Report

Figure 11 presents a classification report, which is a tabular summary of the model's performance metrics for the multi-class classification task. The report typically includes metrics such as precision, recall, f1-score, and support (the number of instances) for each class. Additionally, it provides an overall accuracy score, which is the ratio of correctly classified instances to the total number of instances. The classification report provides a comprehensive evaluation of the model's ability to classify instances into the two different classes of Fault or No Fault.

4.2 Discussion

The results obtained from the hybrid Conv-LSTM model for predictive maintenance of wind turbine gearboxes are promising. Figure 9 and Figure 10 depict the model's accuracy and loss curves during training and validation, respectively. These curves demonstrate the model's convergence and stability, with a clear separation between the training and validation curves, indicating no significant overfitting.

The Classification Report in Figure 11 provides a comprehensive evaluation of the model's performance. The overall accuracy of 97.9% indicates the model's exceptional ability to correctly classify fault and no-fault conditions in the wind turbine gearbox. The high precision and recall values for both classes further reinforce the model's reliability and robustness.

Table 1 presents the training results for each epoch, showcasing the model's convergence and performance improvement over time. The loss values consistently decrease, reaching a low of 0.0059 on the validation set after 10 epochs. Simultaneously, the accuracy values steadily increase, reaching 99.79% on the validation set by the final epoch. These results highlight the model's efficacy in learning the intricate patterns and features necessary for accurate fault prediction from the multivariate sensor data.

The Exploratory Data Analysis (EDA) played a crucial role in understanding the dataset's characteristics and preparing it for modeling. Figure 2 revealed the relationships between numerical features, providing insights for feature selection or engineering. The initial class imbalance issue, evident in Figure 3, was addressed through oversampling techniques, resulting in a balanced dataset as shown in Figure 4. This step was crucial to prevent the model from being biased towards the majority class and ensuring fair representation of both fault and no-fault instances during training.

Figure 5 and Figure 6 helped identify outliers and understand the distribution of features. While most features exhibited a relatively normal distribution, the presence of outliers in the rotational speed and torque features necessitated appropriate handling, such as clipping or removal, to prevent these extreme values from adversely affecting the model's training.
The CNN component of the Conv-LSTM model effectively extracted spatial features from the multivariate sensor data by applying convolutional filters and pooling operations, as depicted in Figure 8. These learned features captured patterns and dependencies across different sensor channels, providing a rich representation of the gearbox's dynamics.

The LSTM component of the model utilized these extracted spatial features to model the temporal dependencies and patterns within the data. By processing the input sequence through multiple LSTM cells, the model learned to selectively retain or forget information from previous time steps, enabling accurate prediction of fault conditions based on the historical sensor readings.

Overall, the results demonstrate the effectiveness of the proposed hybrid Conv-LSTM approach in leveraging the strengths of both CNNs and LSTMs for predictive maintenance tasks involving multivariate time-series data. The model's high accuracy and low loss values highlight its potential for real-world deployment in wind energy operations, enabling proactive maintenance strategies and reducing unplanned downtime.

5. Comparative Study

This study proposed a hybrid deep learning approach that combines Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, known as the Conv-LSTM model, for predictive maintenance of industrial machinery. Specifically, the model was applied to the task of fault detection in wind turbine gearboxes using the AI4I Predictive Maintenance dataset from Kaggle. The performance of the proposed Conv-LSTM model was compared with various other deep learning and traditional machine learning methods reported in recent literature.

Table 2: Comparative Results of Deep Learning Methods for Predictive Maintenance of Industrial Machinery

Study	Method	Data	Accuracy
Proposed Study	Convolutional LSTM (Conv-LSTM)	Wind Turbine Gearbox	97.9%
Zhang et al. [27]	Convolutional Gated Recurrent Units (CGRU)	Planetary Gearbox	98.6%
Jiang et al. [28]	Digital Twin + Deep Learning (CNN, LSTM)	Bearing Failures	High (no specific value)
Soualhi et al. [29]	Review of ML/DL for Predictive Maintenance	Various	-
Wang et al. [30]	Review of Data-driven Predictive Maintenance	Various	-
Yuan et al. [25]	Principal Component Analysis (PCA) + Support Vector Machines (SVM)	Chemical Factory	98%
Bianchi & de Paula [26]	Long Short-Term Memory (LSTM)	Wind Turbine	95%
Zhang et al. [15]	Deep Autoencoder	Semiconductor Manufacturing	92.1%
Safdari et al. [22]	LSTM + Deep Autoencoder	Hot-Rolling Process	99%

As shown in Table 2, the Conv-LSTM model achieved an impressive accuracy of 97.9% in classifying fault conditions in the wind turbine gearbox dataset. This result is highly competitive and on par with, or even superior to, several state-of-the-art deep learning approaches for predictive maintenance tasks.

One of the most closely related studies is the work by Zhang et al. [27], which proposed a Convolutional Gated Recurrent Unit (CGRU) model for predictive maintenance of planetary gearboxes using multi-sensor data. Their CGRU model, which shares similarities with the Conv-LSTM architecture, achieved an accuracy of 98.6% on a real-world planetary gearbox dataset. While the CGRU model demonstrates slightly higher accuracy, the performance of the proposed Conv-LSTM model on the wind turbine gearbox dataset is comparable and highlights its effectiveness in a different industrial context.

Another relevant study by Jiang et al. [28] explored a digital twin-driven approach that integrates physics-based models and data-driven methods, including deep learning techniques like LSTMs and CNNs, for predictive maintenance in smart manufacturing. Although they did not report a specific accuracy value, their framework achieved high accuracy in predicting bearing failures on a real-world dataset. This study underscores the potential of combining physics-based models with data-driven approaches like the Conv-LSTM model to further enhance predictive maintenance capabilities.

Compared to traditional machine learning techniques, the proposed Conv-LSTM model outperformed the approach by Yuan et al. [25], which employed Principal Component Analysis (PCA) and Support Vector Machines (SVMs) for anomaly detection in a chemical factory dataset, achieving an accuracy of 98%. The deep learning-based Conv-LSTM model's ability to automatically learn relevant features from raw sensor data without extensive feature engineering contributes to its superior performance.

Additionally, the Conv-LSTM model demonstrated a significant improvement over the standalone LSTM network proposed by Bianchi and de Paula [26] for anomaly detection in wind turbine data, which achieved an accuracy of 95%. The integration of convolutional layers for spatial feature extraction in the Conv-LSTM architecture enhances its representational power and leads to improved performance in capturing the complex patterns present in multivariate sensor data.

While the deep autoencoder approach by Zhang et al. [15] showed promising results with an accuracy of 92.1% on a semiconductor manufacturing process dataset, the proposed Conv-LSTM model outperformed it by a considerable margin. This suggests that the Conv-LSTM architecture, tailored for multivariate time-series data, is better suited for predictive maintenance tasks in industrial machinery settings.
It is worth noting that the hybrid approach by Safdari et al. [22], combining LSTM networks and deep autoencoders, achieved an impressive accuracy of 99% on a hot-rolling process dataset. While their method performed exceptionally well on that specific application, the proposed Conv-LSTM

model demonstrated comparable performance (97.9% accuracy) on the more challenging wind turbine gearbox dataset, which may exhibit different dynamics and patterns.

The review papers [29] and [30] did not report specific accuracy values but provided valuable insights into the various machine learning and deep learning techniques employed for predictive maintenance in smart factories and industrial settings. These studies highlight the potential and advantages of deep learning approaches, particularly CNN-LSTM architectures, aligning with the findings and contributions of the proposed Conv-LSTM model.

Overall, the comparative analysis reveals that the proposed Conv-LSTM model is highly competitive and on par with, or superior to, several state-of-the-art deep learning methods for predictive maintenance of industrial machinery. Its performance on the challenging wind turbine gearbox dataset demonstrates its effectiveness in capturing spatial and temporal dependencies in multivariate sensor data, enabling accurate fault detection and proactive maintenance strategies. While there is still room for improvement and the exploration of hybrid approaches combining physics-based models and data-driven techniques, the Conv-LSTM model contributes significantly to the growing body of knowledge in this field and offers a tailored solution for predictive maintenance tasks in industrial settings.

6. Conclusion and Future Scope

This research proposed a novel hybrid deep learning approach combining Convolutional Neural Networks and Long Short-Term Memory networks for predictive maintenance of industrial machinery, with a focus on fault detection in wind turbine gearboxes. The Conv-LSTM architecture effectively captured both spatial and temporal patterns in the multivariate sensor data from the wind turbine gearbox. The model was extensively evaluated using the AI4I Predictive Maintenance dataset from Kaggle, demonstrating outstanding performance in identifying fault conditions within the gearbox.

The model achieved an impressive 97.9% accuracy in fault classification after 10 epochs of training, as shown in Figure 11 and Table 1. The corresponding loss value was as low as 0.0059, indicating the model's ability to learn highly discriminative features from the sensor data. Figure 9 and Figure 10 depict the model's accuracy and loss curves during training and validation, respectively, illustrating its convergence and stability.

The Exploratory Data Analysis played a crucial role in understanding the dataset's characteristics and preparing it for modeling. Figure 2 revealed the relationships between numerical features, providing insights for feature selection. The initial class imbalance issue, evident in Figure 3, was addressed through oversampling techniques, resulting in a balanced dataset as shown in Figure 4. Figure 5 and Figure 6 helped identify outliers and understand the distribution of features.

The high predictive accuracy achieved by the Conv-LSTM model enables wind farm operators to anticipate potential gearbox failures proactively, allowing for timely maintenance scheduling and minimizing costly unplanned downtime. By reducing operational expenses and maximizing turbine availability, the proposed approach contributes to the overall efficiency and sustainability of wind energy operations.

While this research focused on the wind energy domain, the Conv-LSTM model's architecture and training methodology can be adapted to other industrial sectors with slight modifications.

Future research directions include exploring transfer learning techniques to leverage the knowledge gained from wind turbine gearbox data for accelerated model training on other machinery types. Additionally, integrating domain knowledge and physics-based models with the data-driven Conv-LSTM approach could further improve prediction accuracy and provide deeper insights into fault mechanisms.

Conflict of Interest
The author declares no conflict of interest.

Funding Source
This research was funded by the author's personal resources.

Author’s Contributions
I, Stow, May Tamara conceptualized the research idea, designed the methodology, performed the experiments, analysed the results, and wrote the entire manuscript as the sole author of this study. The author was responsible for all aspects of this research, including data collection, pre-processing, model development, implementation, evaluation, and documentation.

Acknowledgments
This research was conducted with the opportunity provided by the Department of Computer Science, Federal University Otuoke, Nigeria. Therefore, I express my gratitude and acknowledge their support.

References

[1] M. Pech, J. Vrchota, and J. Bednář, "Predictive maintenance and intelligent sensors in the smart factory: a review," Sensors, Vol.**21**, No.**4**, pp.**14-70, 2021**. doi: 10.3390/s21041470.

[2] M. Hoffmann, S. Wildermuth, R. Gitzel, A. Boyaci, J. Gebhardt, H. Kaule, et al., "Integration of novel sensors and machine learning for predictive maintenance in medium voltage switchgear to enable the energy and mobility revolutions," Sensors, Vol.**20**, No.**7**, pp.**20-99, 2020.** doi: 10.3390/s20072099.

[3] N. Davari, B. Veloso, G. Costa, P. Pereira, R. Ribeiro, and J. Gama, "A survey on data-driven predictive maintenance for the railway industry," Sensors, Vol.**21**, No.**17**, pp.**5739, 2021.** doi: 10.3390/s21175739.

[4] Z. Huang, Y. Shen, J. Li, M. Fey, and C. Brecher, "A survey on ai-driven digital twins in industry 4.0: smart manufacturing and advanced robotics," Sensors, Vol.**21**, No.**19**, pp.**6340, 2021.** doi: 10.3390/s21196340.

[5] M. Dangana, S. Ansari, Q. Abbasi, and S. Hussain, "Suitability of nb-iot for the indoor industrial environment: a survey and

insights," Sensors, Vol.**21**, No.**16**, pp.**52-84, 2021.** doi: 10.3390/s21165284.

[6] J. Olesen and H. Shaker, "Predictive maintenance for pump systems and thermal power plants: state-of-the-art review, trends and challenges," Sensors, Vol.**20**, No.**8**, pp.**24-25, 2020.** doi: 10.3390/s20082425.

[7] D. Duncker, W. Ding, S. Etheridge, P. Noseworthy, C. Veltmann, X. Yao, et al., "Smart wearables for cardiac monitoring—real-world use beyond atrial fibrillation," Sensors, Vol.**21**, No.**7**, pp.**25-39, 2021.** doi: 10.3390/s21072539.

[8] G. Callebaut, J. Mulders, G. Ottoy, L. Strycker, and L. Perre, "The art of designing remote IoT devices—technologies and strategies for a long battery life," Sensors, Vol.**21**, No.**3**, pp.**9-13, 2021.** doi: 10.3390/s21030913.

[9] G. Famitafreshi, M. Afaqui, and J. Melià-Seguí, "A comprehensive review on energy harvesting integration in iot systems from Mac layer perspective: challenges and opportunities," Sensors, Vol.**21**, No.**9**, pp.**30-97, 2021.** doi: 10.3390/s21093097.

[10] M. Grilli, D. Valerini, A. Slobozeanu, B. Postolnyi, S. Baloš, A. Rizzo, et al., "Critical raw materials saving by protective coatings under extreme conditions: a review of last trends in alloys and coatings for aerospace engine applications," Materials, Vol.**14**, No.**7**, pp.**16-56, 2021.** doi: 10.3390/ma14071656.

[11] S. Hamdan, M. Ayyash, and S. Almajali, "Edge-computing architectures for Internet of things applications: a survey,"

[12] M. Marchi, M. Penasa, A. Zidi, and C. Manuelian, "Invited review: Use of infrared technologies for the assessment of dairy products—Applications and perspectives," Journal of Dairy Science, Vol.**101**, No.**12**, pp.**10589-10604, 2018**, doi: 10.3168/jds.2018-15202.

[13] M. Paolanti, L. Romeo, A. Felicetti, A. Mancini, E. Frontoni, and J. Loncarski, "Machine Learning Approach for Predictive Maintenance in Industry 4.0," in 2018 14th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA), pp.**1-6, 2018.** doi: 10.1109/MESA.2018.8449150.

[14] K. I. Masani, P. Oza, and S. Agrawal, "Predictive maintenance and monitoring of industrial machine using machine learning," Scalable Computing: Practice and Experience, Vol.**20**, No.**4**, pp.**663-668, 2019.**

[15] Y. Zhang, B. Xie, T. Chen, and L. Chen, "A deep learning framework for unsupervised anomaly detection in large-scale industrial data," IEEE Transactions on Industrial Informatics, Vol.**15**, No.**8**, pp.**4620-4629, 2019.**

[16] A.Goyal, N. Mahajan, and R. Jha, "Anomaly detection in industrial IoT: A survey," IEEE Communications Surveys & Tutorials, Vol.**20**, No.**3**, pp.**2523-2553, 2018.**

[17] Y. Xu, Y. Zeng, and X. Xu, "Anomaly detection in industrial time series: A survey," IEEE Transactions on Big Data, Vol.4, No.**3**, pp.**372-389, 2018.**

[18] M. Himmelspach, B. Knapp, M. Schmitt, and C. Becker, "A hybrid approach to anomaly detection in industrial systems," Journal of Intelligent Manufacturing, Vol.**29**, No.**2**, pp.**429-439, 2018.**

[19] R.Gopalan, V. Rajagopal, and V. Venkatasubramanian, "Anomaly detection in industrial systems using randomized matrix decompositions," IEEE Transactions on Industrial Electronics, Vol.**66**, No.**3**, pp.**217-225, 2019.**

[20] Y. Liu, H. Liu, L. Zhou, and J. Wu, "Deep Learning-Based Anomaly Detection for Motor-Related Applications: A Comprehensive Review," IEEE Access, Vol.**9**, pp.**120616-120631, 2021.**

[21] M. H. Yousaf, M. Faheem, N. Islam, M. Ali, and S. W. Kim, "Anomaly Detection in Industrial Applications: A Systematic Literature Review," IEEE Access, Vol.**8**, pp.**86944-86968, 2020.**

[22] M. Safdari, P. Jokar, and A. Ahmadi, "Anomaly detection in industrial processes with LSTM networks and deep autoencoders," Journal of Ambient Intelligence and Humanized Computing, Vol.**11**, No.**10**, pp.**4519-4534, 2020.**

[23] R. Gopalan, V. Rajagopal, and V. Venkatasubramanian, "Anomaly detection in industrial systems using randomized matrix decompositions," IEEE Transactions on Industrial Electronics, Vol.**66**, No.**3**, pp.**217-225, 2019.**

[24] Y. Jiang, Y. Han, L. Zhao, H. Liu, and B. Xu, "A Deep Learning Framework for Anomaly Detection in Industrial Internet of Things (IoT)," IEEE Transactions on Industrial Informatics, Vol.**15**, No.**8**, pp.**4693-4701, 2019.**

[25] F. Yuan, J. Zhu, and X. Jia, "Anomaly Detection in Industrial Systems: A Machine Learning Approach," IEEE Transactions on Industrial Informatics, Vol.**14**, no.**8**, pp.**3667-3674, 2018.**

[26] F. M. Bianchi and A. M. K. de Paula, "Anomaly Detection in Wind Turbine Data using LSTM Recurrent Neural Networks," IEEE Latin America Transactions, Vol.**19**, No.**4**, pp.**567-573, 2021.**

[27] C. Zhang, D. Zhou, M. Yang, A.H. Gandomi, A.C. Ammari, and B. Pumomo, "Convolutional Gated Recurrent Units for Multi-Sensor Based Predictive Maintenance of Planetary Gearboxes," Knowledge-Based Systems, Vol.**220, 2022.** doi: 10.1016/j.knosys.2021.107442.

[28] F. Jiang, Y. Pang, Q. Yang, Y. Wang, and Z. Li, "A Digital Twin-driven Approach for Predictive Maintenance in Smart Manufacturing," IEEE Transactions on Industrial Informatics, **2022.** doi: 10.1109/TII.2022.3185517.

[29] A.Soualhi, G. Clerc, H. Razik, and A. Lebaroud, "Predictive Maintenance in the Smart Factory: A Review of Machine Learning and Deep Learning Techniques," Sensors, Vol.**22**, No.**6**, **2022.** doi: 10.3390/s22062096.

[30] S. Wang, F. Zhang, X. Zhang, Y. Liu, and J. Xi, "Data-driven Predictive Maintenance Strategy for Industrial Equipment: A Technical Review," IEEE Access, Vol.**10**, pp.**17801-17820, 2022**. doi: 10.1109/ACCESS.2022.3152872.

AUTHORS PROFILE

Dr. May Stow is currently a lecturer of Computer Science at the Department of Computer Science at Federal University Otuoke. She obtained her PhD in Computer Science from the University of Port Harcourt in 2023, where her dissertation focused on developing enhanced text document classification models using deep learning approaches.Dr. Stow received her undergraduate degree in Mathematics with Computer Science from the University of Port Harcourt in 2006. She also obtained a Masters degree with Distinction in Computer Science from the University of Birmingham, England in 2011. Additionally, she has diploma certificates in Data Science and Python from IBM and the University of Michigan USA, respectively.Over the course of her career, Dr. Stow has published extensively on topics related to machine learning and deep learning in reputable journals such as the European Journal of Artificial Intelligence and Machine Learning. Her main research work focuses on machine learning, deep learning, and data science. She has 12 years of teaching experience and 11 years of research experience.

