International Journal of Research and Technology (IJRT)
International Open-Access, Peer-Reviewed, Refereed, Online Journal
ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529
| An ISO 9001:2015 Certified Journal |

IoT-Driven Predictive Maintenance Model for Rotating Machinery Using Machine Learning and Deep Learning Techniques

[1]**Surendra Singh Bisht**

Research Scholar, **Department** of Computer Science and Engineering (CSE), Om Sterling Global University Hisar, India

sbisht2000@gmail.com, surendracse221@osgu.ac.in

[2]**(Dr.) Saurabh Charaya**

Professor, School of Engineering and Technology, Om Sterling Global University

Hisar, India

saurabh.charaya@gmail.com

[3]**Dr. Rachna Mehta**

Associate Professor, School of Engineering and Technology,

Om Sterling Global University, Hisar, India

drrachnamehta@osgu.ac.in

Abstract- In modern factories, it is very important to make sure that equipment problems are found quickly and accurately so that operations stay efficient and costly downtime is avoided. The H-CBLSTMNet is a novel and better technique to discover problems with bearings for health monitoring. The model learns in order with Bidirectional LSTM layers and finds spatial characteristics with Convolutional Neural Networks. This helps it automatically pick up on both short-term and long-term patterns in raw sensor data. The proposed model outperformed traditional machine learning and independent deep learning models, achieving a test accuracy of 98.73%, with Precision, Recall, and F1-Score metrics of 98.74%, 98.73%, and 98.74%, respectively. These results suggest that the classification is fair, with very few false positives and negatives. A high training accuracy of 97.80% and a high validation accuracy of 96.83%, along with a low test loss of 0.0436, showed that strong generalization was possible. Using Adam optimization, early halting, and dropout regularization made this achievable. The confusion matrix showed that the faults were divided into several groups, which proved that the diagnostic performance was accurate. Models like SVM, KNN, and Random Forest, on the other hand, have a hard difficulty discovering features that are very different from each other. In general, H-CBLSTMNet is a mechanism to find bearing problems in real time that can be scaled and automated. It cuts down on manual feature engineering and promotes Industry 4.0 predictive maintenance by making machines more reliable, letting people make decisions ahead of time, and reducing down on downtime.

Keyword: Vibration Signal Processing, Predictive Maintenance, Bearing Fault Diagnosis, Industry 4.0, Condition Monitoring, Robust

1. Introduction

The Industrial Internet of Things has grown very quickly and revolutionized the way things are created now. This has allowed firms to move from reactive and preventative maintenance to very successful predictive maintenance solutions. In sectors including manufacturing, energy, aircraft, automobiles, and process, things work because of rotating devices like bearings, motors, compressors, turbines, pumps, and gearboxes. If these parts break, it might be very expensive to fix them, make things dangerous, and stop working [1]. Old ways of keeping matters up to date are either based on time or impulsively, which can lead to extra work or unforeseen malfunctions. Because of this, more and more businesses are using IoT-driven predictive maintenance models that employ Machine Learning and Deep Learning to find problems as they happen, make operations more reliable, and help people make better choices. IoT and PdM work together to make it possible to employ sensors to keep an eye on equipment all the time and acquire real-time information on things like vibration, temperature, noise emissions, motor current indicators, and pressure [2]-[4]. Edge and cloud computing help send, store, and evaluate these huge amounts of data from sensors so that problems can be found early. Random Forest, Support Vector Machine (SVM), K-Nearest Neighbor (KNN), and Logistic Regression are all examples of machine learning approaches that have been used a lot to sort out different forms of faults and health problems in rotating machinery. But ML-based systems depend a lot on hand-crafted features, which means they can't find hidden and complicated defect patterns in raw data very well [5].

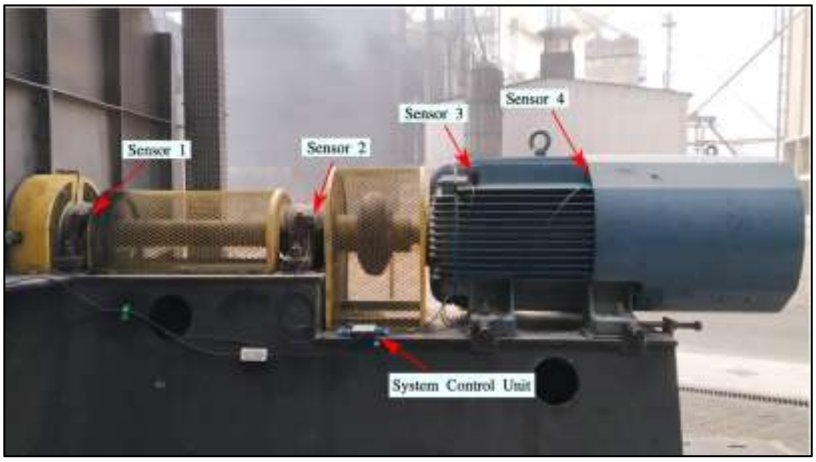

Figure 1 Rotating Machinery [6]

Recent improvements in Deep Learning have made it much easier to find faults by allowing automated feature extraction directly from raw time-series or frequency-domain data. Deep architectures, including Convolutional Neural Networks, Recurrent Neural Networks, Long Short-Term Memory networks, and hybrid models, have demonstrated exceptional precision in acquiring both spatial and temporal representations of machinery action. CNNs are good at picking out important features from raw signals, while LSTMs and BiLSTMs are good at learning how things change over time and how faults progress over time. IoT-driven gathering of data and DL-based cognitive computation have permanently transformed predictive maintenance [7]-[10]. Now, businesses can spot problems early on and guess how they will get worse before they cause big problems. Four primary parts make up a conventional IoT-enabled predictive maintenance system: (i) Sensor-based data collection, where multiple sensors collect operational data; (ii) data transmission and storage, which uses industrial IoT gateways, edge devices, and cloud platforms; (iii) data processing and fault diagnosis, which uses ML/DL models to find fault patterns; and (iv) maintenance decision-making, which gives useful information for planning maintenance. Continuous tracking and sophisticated data analysis can be used to do condition-based maintenance. This helps in running things smoother, reduces on the downtimes and increases the lifespan of machines. [11].

Deep-learning predictive maintenance is now a highly accurate and scalable method that can find application in Industry 4.0. DL models are more efficient in maintaining a watch on the health of rotating machine since they are capable of managing noisy data, nonlinear signal changes, and complex fault states. Traditional procedures don't work this way. Hybrid models that mix CNN and LSTM or attention-based architectures are also garnering a lot of attention since they work better at capturing the multi-dimensional features of vibration signals [12]-[14]. A lot of progress has been achieved, there are still problems to tackle when it comes to building predictive maintenance models that use the Internet of Things (IoT). These include working with huge amounts of data that have many dimensions, making sure that data is safe, allowing edge devices to make inferences in real time, and improving models' performance in scenarios they haven't seen before. We need to perform research and come up with new ideas to make models stronger, utilize less processing resources, and build frameworks that can be used in business that can grow. Using Machine Learning and Deep Learning in IoT-based predictive maintenance is a revolutionary technique to keep an eye on the condition of rotating machinery. Industries can reduce the number of unexpected failures, improve maintenance schedules, and make machines function better overall by using real-time networks of sensors, smart data analytics, and automated fault detection. This will ensure that the manufacturing systems become more intelligent, safer, and less expensive and at the same time satisfy the requirements of Industry 4.0 [15]-[20].

2. Literature Review

Emmanuel 2025 et.al AI has transformed maintenance planning (PdM) by assisting companies in determining when their equipment will malfunction and simplifying the process of upkeep by the company. Machine learning, deep learning, and systems that are run by IoT collaborate to simplify predictive maintenance in various industries, including manufacturing, steel, and information and communication technology (ICT). The research study discusses necessary steps of the process such as obtaining data, cleaning, building a model, and decision making. It also discusses ways in which PdM can save money, reduce downtime and make it safer. Even with the benefits, the quality of the data, the correctness of the model, and the preparation of the organization continue to be a question even though the model has

International Journal of Research and Technology (IJRT)
International Open-Access, Peer-Reviewed, Refereed, Online Journal
ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529
| An ISO 9001:2015 Certified Journal |
certain advantages. The article explains how AI can be used to improve Industry 4.0 through the use of data to improve maintenance [21].
Ūselis 2025 et.al Discuss the issues of the implementation of predictive maintenance in the case of inexpensive edge devices with minimal memory, computing capability, or battery life. The research paper implements an RP2040 microprocessor to investigate the vibration and motor current signals to identify defects in areas where resources are constrained. We applied the time-domain statistical features and PCA as a means of improving the diagnosis, followed by XGBoost classification. Model distillation was less memory-consuming without slowing down which facilitated finding errors on the device. The vibration data and current were also used with an accuracy of 94.1 and 95.5 percent respectively. The outcome is that it is possible to use inexpensive embedded systems to conduct condition monitoring with excellent accuracy. This demonstrates that PdM can be employed by companies that are concerned with costs [22].
Abdalah 2025 et.al underline the necessity of predictive maintenance in Industry 4.0 that is being propelled by the necessity of IIoT systems to identify and fix the faults as soon as possible. The experiment gathered a large amount of the data of three AC motors with vibration, current, and temperature sensors and running under different faulty conditions. This built a custom dataset. The cloud taught a seven-layer Deep Neural Network (DNN) the way to categorize things. The model was actually accurate and did not lose much hence it was good as both training and testing. The study demonstrates that real-time Predictive Maintenance (PdM) based on the use of IoT technology will be applicable to detect issues, reduce the probability of failures, and enable proactive maintenance in the industrial environment. [23].
Alhuqay 2024 et.al explore how machine learning and IIoT can be combined to enhance predictive maintenance at factories. The study evaluates four machine learning models to determine the most suitable one in terms of equipment health monitoring and predicting failures. It states that systems require not only routine maintenance but also data-driven insights. The analysis demonstrates that the selection of the appropriate model is extremely important to a successful PdM rollout. The findings inform us of so much about the working of IIoT and ML in synergy. This assists businesses in reducing downtime, maintaining their assets in better conditions, as well as, operating their businesses in a more efficient manner. [20].
Haque 2024 et.al follow PRISMA criteria with the intention of a complete survey of 78 articles about AI-driven predictive maintenance. The review proves that CNN and LSTM, machine learning and deep learning models are much more accurate in their predictions, they reduce maintenance expenses and the equipment works longer. There is even better with the digital twin technology which engages with the Internet of Things and real-time monitoring. The paper similarly discusses the significance of edge and cloud computing to develop platforms of predictive maintenance, which can expand and evolve. Nevertheless, such aspects as the ease of comprehending models, the quality of the data and cybersecurity are

still problematic. The evaluation demonstrates that PdM may be a lengthy and effective means to automatise industries [24].

Table 2.1 Literature Summary

Author & Year	Methodology / Technique	Research Gap	Key Findings	Limitations
Hanifu 2025 [25]	Vibration & temperature data Preprocessing using Isolation Forest; Predictive modeling using ARIMA, RF & LSTM with SMBO-TPE	Limited integration of IoT sensor-based vibration & thermal data into a unified ML-based predictive model for industrial motors	ARIMA performed best for predicting vibration trends; Residual analysis effectively detected anomalies & early fault symptoms;	Focused mainly on time-series forecasting, not deep learning hybrid models; Limited to one factory environment, reducing generalizability
Luu & Huynh (2025) [26]	ResNet-based autoencoder for feature extraction; Deep Reinforcement Learning using Soft Actor-Critic (SAC) for dynamic RUL prediction	Lack of models that learn RUL through interaction rather than static regression; poor adaptability in existing models to different degradation conditions	SAC-ResNet model improved RUL prediction accuracy and adaptability; Demonstrated superior performance on PHM 2012 dataset compared to traditional regression approaches	Tested only on one benchmark dataset; Requires high computational cost and RL expertise for implementation
R & Mutra (2025) [27]	Vibration data analysis under multiple fault conditions; EEMD-based feature extraction; ML models—RBFNN, BPNN, CPNN; ANFIS for classification	Limited studies classifying multiple TRB fault types simultaneously using hybrid neuro-fuzzy models	ANFIS achieved 90.9% accuracy for multi-fault TRB diagnosis; EEMD effectively extracted meaningful features linked to specific fault types	Accuracy below modern DL benchmarks; Lab-based conditions—may not reflect real industrial noise and variability
Ovie Vincent	IoT, AI, and ML-enabled Predictive	Lack of consolidated	PdM significantly reduced	High sensor deployment cost,

International Journal of Research and Technology (IJRT)
International Open-Access, Peer-Reviewed, Refereed, Online Journal
ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529
| An ISO 9001:2015 Certified Journal |

Erhueh et al. (2024) [28]	& Condition-Based Maintenance for rotating machinery; Real-time monitoring using vibration,	insights on real-world PdM deployment challenges & lessons in energy sector machinery	downtime, increased reliability & optimized maintenance costs in energy systems	data integration challenges & need for skilled workforce restrict large-scale implementation
Magadan et al. (2023) [29]	TabPFN model for fault classification using limited RM dataset; Compared with XGBoost & RF on three public datasets	Limited availability of real RM fault datasets for training DL/ML models; Need models that perform well with small datasets	TabPFN outperformed conventional models with limited samples & ideal for real-industry deployments with small data	Performance depends on GPU availability; Does not address temporal feature learning needed for sequential vibration data

3. Research Methodology

The next part explains the research strategy used to develop a predictive maintenance system for rotating machinery using machine learning and vibration data from the Internet of Things It explains about the study plan, where the information originated from, what the dataset is like, how the data was cleaned up, how the parameters were produced, how the model was made, how the efficacy was tested, and what tools were utilized. The methodological framework is there to make sure that the predictive maintenance system works correctly is dependable and can be used again.

3.1 Research Design

The study combines an experimental and analytical research methodology, utilizing data-driven machine learning and deep learning approaches to detect bearing issues. The technique include the acquisition of IoT-based vibration data, the preparation and preprocessing of the dataset, the extraction of time and frequency domain characteristics, the development of machine learning/deep learning models, and the assessment of their performance. The method is quantitative and uses model comparison to find the optimum predictive maintenance model for use in industry.

3.2 Data Collection

The Case Western Reserve University (CWRU) Bearing Dataset, which was downloaded from Kaggle, was used in this study. It has real-time vibration signals from electric motor bearings in several health states: normal, inner race fault, outer race fault, and ball fault, with fault widths of 0.007", 0.014", and 0.021". At the Drive End (DE) and Fan End (FE), accelerometers recorded data at different motor loads (0–3 HP). The .mat files have vibration signals in the time domain, which makes them good for machine failure diagnosis research. Even while this study employs a publicly accessible benchmark dataset, the CWRU data is like genuine IoT-based industrial settings where vibration sensors constantly send machine health data. In steel mills, comparable sensors on rotating machinery (such motors, bearings, and gears) provide vibration signals through IoT gateways to keep an eye on the condition of the equipment. From https://www.kaggle.com/datasets/brjapon/cwru-bearing-datasets So, what we learn from the CWRU dataset can be employed in IoT-enabled predictive maintenance systems in steel mills.

3.3 Dataset Preparation Procedure

1. The process of preparing the dataset was very important for turning the raw vibration signals into a structured form that could be used for machine learning. This step made sure that the data from the Case Western Reserve University Bearing Dataset was properly labeled, processed, and split up so that the model could be trained and tested properly. The preparation process was done in a systematic four-step way. The first step, Label Extraction, was to look at the names of the .mat files and find out what kind of problem they were. The file names had information about the condition of the bearing, which made it feasible to automatically arrange specimens into four groups: normal condition, inner race fault, outer race fault, and ball fault. Some types of text were assigned numbers in order to interact with each other. Signal Loading was the second stage which consisted of scipy.io library to access vibration signals. The Drive-End data of sensors capture more attention on our part because they are nearer to a bearing and thus can identify issues earlier. Signal Segmentation phase (3) attempts to subdivide large and continuous vibration patterns into smaller interconnected intervals. This division of the data into half gave us more samples that we could train on, and we ensured that both sides had sufficient time information to demonstrate various forms of error. The model also learnt more effectively when it possessed additional information on the points. At the fourth step, Dataset Construction, all the processed sections were combined in to a structured data set, which had two primary components: X, the characteristic matrix containing separated signals, and y, the encoded labels. This made it possible to build a full and balanced dataset that could be used to educate machine learning and deep learning models how to do predictive maintenance.

3.4 Data Preprocessing

When working with vibratory machine learning datasets, data pretreatment is very important since it ensures sure that the data is all the same, consistent, and good for training models. The goal of this stage was to change the dataset that was created into a format that learning techniques could easily understand and use. The preprocessing procedure had three main steps, which are listed below:

2. Label Encoding: This was the first step that we undertook and we used the LabelEncoder function to transform the category class labels into integers. Label encoding transformed the four bearing situations of normal, inner race fault, outer race fault and ball fault into numbers in form of codes. This was essential since deep learning and machine learning algorithms require calculation to perform understanding of numbers. With this update, the supervised learning became more effective and the model became easier to understand. It

International Journal of Research and Technology (IJRT)
International Open-Access, Peer-Reviewed, Refereed, Online Journal
ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529
| An ISO 9001:2015 Certified Journal |

also ensured that the representation of the classes was clear and the same categories of faults were used in both the training and test data.

3. The next stage was to divide the dataset into two parts: a test one and a training one. This was in a manner that the performance of the model would be rated fairly. We used an 80:20 split, in which we taught the models on 80 percent of the data and then held back 20 percent of the data to test the ability of the models to estimate new data. In the split, stratification has been used in order to ensure that the labels in the classes of both sets were equally distributed. This ensured that all the types of errors were equally represented and the classes were not too different. This rendered the model more credible and useful within a broader scope of context.
4. Feature scaling: The the last step was to use Standard Scaler in order to normalize the attributes of the input. The scaling of the values of the characteristics was done in a way that the mean was 0 and standard deviation was 1. This standardization was extremely significant to distance-based algorithms such as KNN as well as to speeding up the convergence to deep learning models. It ensured that there was no single characteristics domination of the learning process due to size disparities. This increased the efficiency of training and the model functioning.

3.5 Feature Engineering

It was highly crucial to do feature engineering to make the prediction models function better because vibration signals are naturally high-dimensional, intricate, and non-linear. It was important to extract useful features that capture signal characteristics in order to make the model more precise and trustworthy because raw vibration data alone is not adequate for effective learning. To fix this, feature extraction was done in two main areas: frequency and time. These two areas together explain how machine vibrations act in both normal and broken circumstances.

Time-Domain Features: Nine statistical features were computed for each segmented signal in the time domain to summarize its amplitude and waveform characteristics. Mean (for average vibration level), Standard Deviation, Skewness and Kurtosis (for peak sharpness) were all instances. We utilized Root Mean Square (RMS) to find out how much energy the signal had and Peak-to-Peak to find out how wide the amplitudes were. We used the Crest Factor, Shape Factor, and Impulse Factor to measure how impulsive and how the waveform shape is, which are both particularly sensitive to bearing difficulties. These time-based indicators assist find problems early on because broken parts emit vibration spikes that aren't normal when there are flaws on the surface.

Frequency-Domain Features: The Fast Fourier Transform was used to split time-series data into its spectrum parts so that they could be looked at in the frequency domain. These were four characteristics: The Spectral Energy is a display of the strength of all the frequencies, and the Spectral Centroid is a display of where the most of the frequency energy is. Dominant Frequency represents the most notable frequency of vibration whereas Frequency Standard Deviation represents the spreadness of spectral distribution. These characteristics present dissimilar vibrations of fault conditions, which may be hard to observe when examining the temporal sphere alone. When time and frequency domain data were used, the model was much better in learning and identifying patterns as well as finding errors. This provided a complete insight into machine shaking..

3.6 EDA

The exploratory data analysis was performed before the model was built to understand more about the vibration signals in CWRU dataset including its structure, distribution and characteristics. This approach studied the distribution of classes, interdependence of signals, amplitude variation and duration of raw signals. This assisted us in identifying patterns and unusual items in the information. EDA informs you about the quality of ability to separate features and what to do before doing it, including normalization or segmentation. The data enabled us to determine the important aspects such as the balance of classes and fault specificity. Using features engineering, AI and deep learning to develop maintenance prediction models for rotating machinery was an excellent place to start.

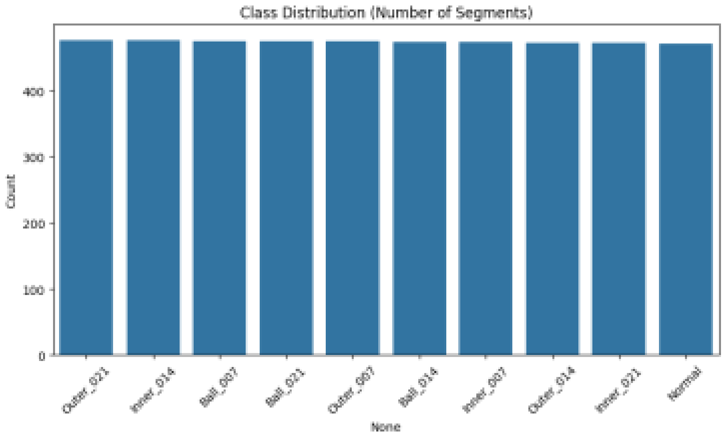

Figure 2 Class Distribution Graph

This figure shows how samples are spread out across different bearing situations in the CWRU dataset. These include normal, ball fault, inner race fault, and outer race fault classes. It shows how many of each class there are in a visual way, which shows whether the dataset is balanced or not.

International Journal of Research and Technology (IJRT)
International Open-Access, Peer-Reviewed, Refereed, Online Journal
ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529
| An ISO 9001:2015 Certified Journal |

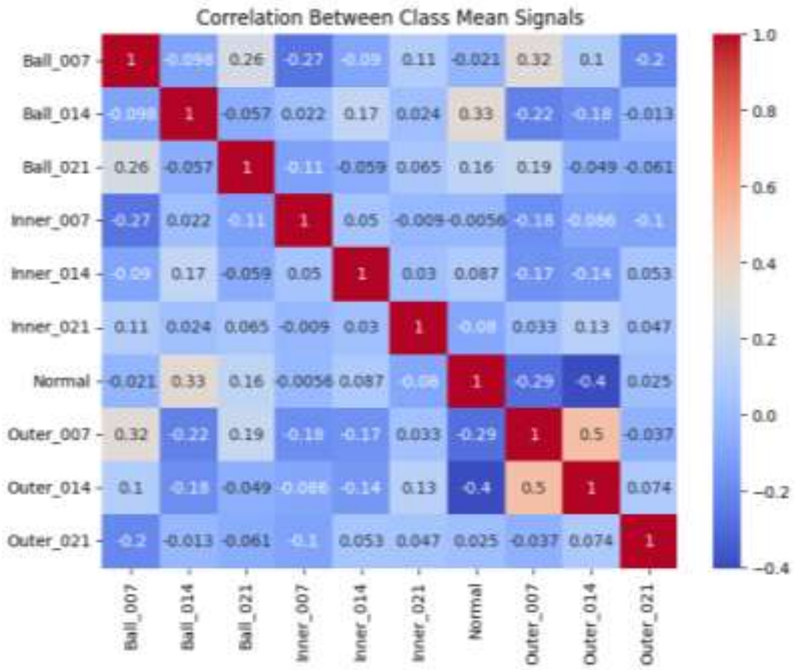

Figure 3 Correlation Between class mean signals

The figure shows how the average vibration signals for each type of bearing condition are connected to each other. The average signals indicate how different types of faults create vibration patterns that are both the same and distinct. A stronger correlation suggests that the signals are similar, whereas a less correlation means that the fault signatures are not. This graphic shows us how to separate features and how well the features we acquired from the data work for training a model. It makes sure that the predictive maintenance model can recognize the difference between normal and bad situations with a high level of accuracy.

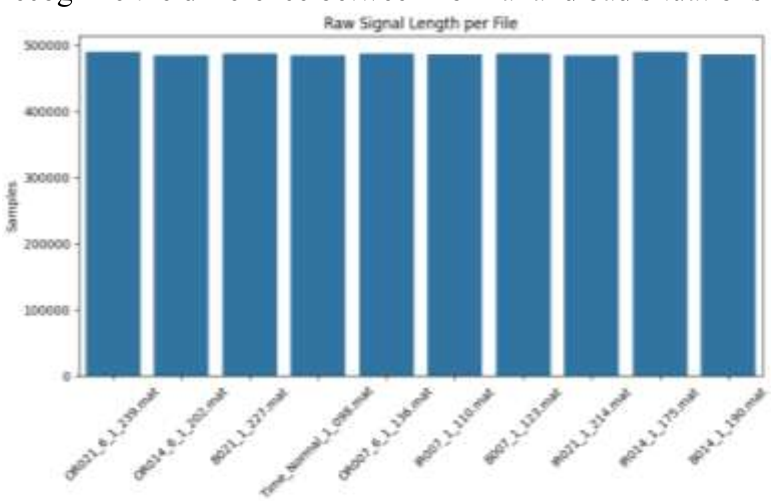

Figure 4 Raw signal Length per file bar Graph

The illustration shows a bar graph that illustrates how long the raw vibration signal sequences are in each one. The CWRU dataset's mat file. Signal durations may vary due to differing sampling periods or motor load circumstances. It's important to know these differences for effective segmentation and model training since uniform or normalized segments make models more consistent. The graph displays how complete the data is and helps you locate files that may need to be changed before they can be used to make sure that each segment has enough information to find faults correctly.

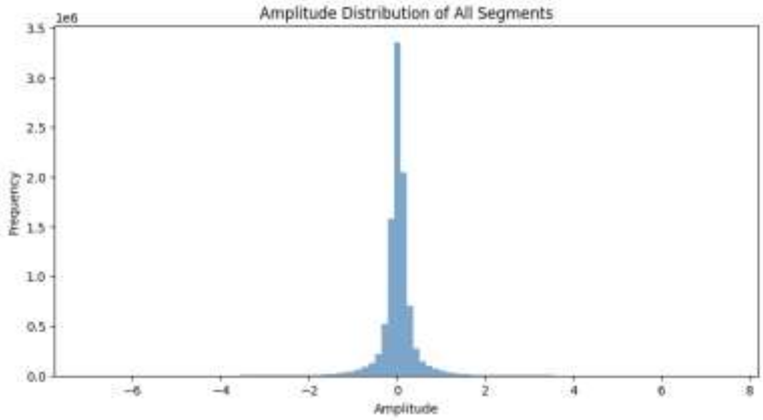

Figure 5 Amplitude Distribution graph

The graphic shows how the amplitude of time-domain vibration signals vary based on the state of the bearing. The graph shows how the intensity of vibrations can alter and how big the range is. It also shows how different patterns are connected to distinct problems, such as ball, inner race, and outside race defects. Peaks and spikes in amplitude are often signals of mechanical difficulties, which are particularly crucial for discovering problems early on. This picture assists with feature selection by displaying which signal qualities can determine the difference between normal and faulty situations in the predictive maintenance model.

.

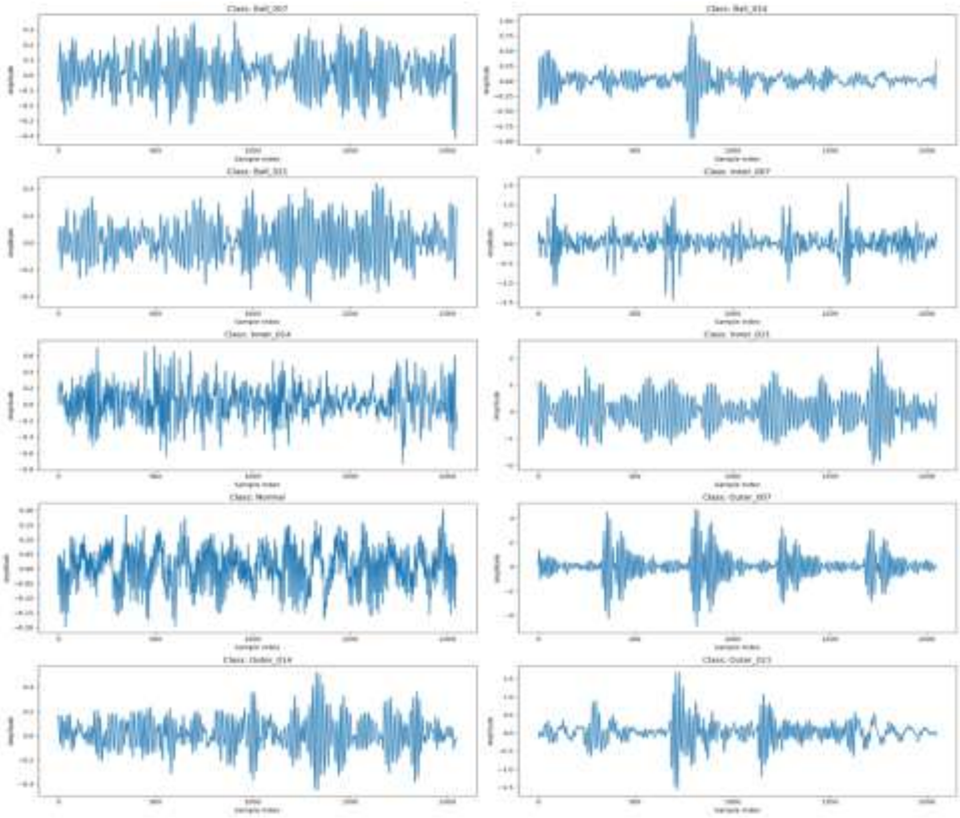

Figure 6 Time-domain vibration signal segments for various bearing conditions

The figure shows the time-domain vibration signal segments for several bearing circumstances, such as Ball, Inner Race, and Outer Race faults (007, 014, 021), as well as normal bearings. The display illustrates how the amplitude changes from one sample index to the next, which makes it easier to spot the patterns that are specific to each type of problem. These variations in the form of the waveform and the amplitude of the signal are noteworthy to identify the problems beforehand.. By examining these components, feature extraction and model training can concentrate on identifying temporal patterns and attributes unique to faults. This will improve the accuracy and reliability of models for predictive maintenance of factory rotating machinery.

3.7 Machine Learning Models

This task involved the creation of machine learning models to categorize bearing situations based on vibration signal features derived from the CWRU dataset. We picked RF and KNN, two well-known algorithms, because they are good at discovering patterns and are strong.

1. Random Forest Classifier

The random forest categorization system is an example of a group of learning systems. While it is training, it makes a number of decision trees and integrates the outcomes to make them stronger and more accurate. This solution makes 200 trees (n_estimators=200), and each one learns from a different set of data and features. This helps reduce the rate of overfitting and change. The model generates its final guesses using the votes of the majority of people. This is very well applied with difficult and non-linear datasets. It is also subjected to measurement

International Journal of Research and Technology (IJRT)
International Open-Access, Peer-Reviewed, Refereed, Online Journal
ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529
| An ISO 9001:2015 Certified Journal |

to ensure that it is applicable to every class. Random Forest is a fabulous tool of sorting as well as determining the most important traits since it is simple to operate and consistent.

2. K-Nearest Neighbors

KNN Classifier is a simple type of learning by doing. It arranges the new samples sequentially, according to the most common label of the samples nearest to them in the feature space. Four or five neighbors (n neighbors=5) are considered in this case and the Euclidean distance is applied to calculate the distance between neighbors. KNN comes in handy when dealing with various types of classification problems, since it does not assume anything about the distribution of the data out. So to view the performance of the model, see its performance and a confusion matrix. KNN is difficult to utilize very large datasets because it requires a great deal of computing time, however, it is simple to comprehend and is suitable in well-clean and well-scaled data.

3.8 Deep Learning Model

The model proposed is an H-CBLSTMNet, which is created using deep learning and applies time-series data which are calculated on the basis of vibrations to determine the extent to which bearings are working. This hybrid architecture learns both spatial and temporal features from IoT sensor data that comes in sequence. It does this by combining the best parts of Convolutional Neural Networks with Bidirectional Long Short-Term Memory (BiLSTM) networks. CNN layers are good at discovering patterns that are only present in a small area and signatures that are only present in a certain type of fault. BiLSTM layers are good at finding long-range temporal dependencies. This allows the model see little differences in how things vibrate when they are working and when they are broken.

The H-CBLSTMNet model consists of three core modules.

A. **CNN Feature Extractor:** This stage extracts local spatial features from 1D vibration signals through three Conv1D blocks. Each block includes a Conv1D layer (filters: 64 → 128 → 256), Batch Normalization for training stability, MaxPooling1D for dimensionality reduction, and Dropout (0.3) to prevent overfitting.
B. **BiLSTM Sequence Modeling:** Two Bidirectional LSTM layers follow for temporal learning. The first BiLSTM (128 units) returns full sequences, while the second (64 units) produces the final temporal representation. Both use Dropout (0.3) to enhance generalization.
C. **Fully Connected Classifier:** The final stage includes a Dense layer (128, ReLU) to learn nonlinear mappings, a Dropout layer (0.3), and a Softmax output layer for multi-class fault classification.

Table 3.1 Hyperparameter table

Component	**Hyperparameter**	**Value**
Model Architecture	Model Type	CNN + BiLSTM
Input Layer	Input Shape	(2048, 1)
Conv1D – Layer 1	Filters	64
	Kernel Size	7

	Activation Function	ReLU
	Padding	Same
	Batch Normalization	Applied
	MaxPooling1D Pool Size	2
	Dropout Rate	0.3
Conv1D – Layer 2	Filters	128
	Kernel Size	5
	Activation Function	ReLU
	Padding	Same
	Batch Normalization	Applied
	MaxPooling1D Pool Size	2
	Dropout Rate	0.3
Conv1D – Layer 3	Filters	256
	Kernel Size	3
	Activation Function	ReLU
	Padding	Same
	Batch Normalization	Applied
	MaxPooling1D Pool Size	2
	Dropout Rate	0.3
BiLSTM – Layer 1	Units	128
	Return Sequences	True
	Dropout Rate	0.3
BiLSTM – Layer 2	Units	64
	Dropout Rate	0.3
Dense – Layer 1	Units	128
	Activation Function	ReLU
	Dropout Rate	0.3
Output Layer	Units	num_classes
	Activation Function	Softmax
Optimizer	Type	Adam
	Learning Rate	1e-4
Loss Function	Type	Sparse Categorical Crossentropy
Metrics	Metrics Used	Accuracy,Precision,Recall,F1
Training Configuration	Epochs	100
	Batch Size	64
	Validation Split	0.1
	Verbose	1

Table 3.1 shows the hyperparameters set up for the proposed hybrid model, H-CBLSTMNet, which is meant to diagnose faults based on vibration signals. The design starts with an input shape of (2048,1). Then, there are three Conv1D layers with 64, 128, and 256 filters and kernel sizes of 7, 5, and 3, respectively. To make sure that feature extraction is strong and overfitting is less likely, each convolutional block employs ReLU activation, equal padding, batch normalization, max-pooling, and a 0.3 dropout rate. Two BiLSTM layers with 128 and 64 units each learn temporal features. There is a thick layer with 128 units and ReLU activation before the softmax output layer. The Adam optimizer (1e-4) trains the model for 100 epochs with a batch size of 64. Accuracy, precision, recall, and F1-score are used to measure performance..

3.9 Model Evaluation Metrics

Five major evaluation indicators were utilized to check how well the suggested model worked and make sure it was powerful and reliable. Accuracy informed us of the overall performance of the classification results in terms of accuracy and Precision informed us of the overall performance of the defect detection by reducing the number of false articles. One of the methods of testing the sensitivity of the model or the ability of the model to identify all the actual problems was through the recall. The harmonic mean of precision and recall (F1-score) helped in the evaluation of the performance of the framework, particularly in cases when classes were not distributed evenly. Mistakes in training and testing were being tracked with the help of loss. This allowed us to monitor the learning of the model and the extent to which the model was able to utilize what it had learned in new circumstances..

3.10 Tools and Technologies Used

The work under consideration utilized a number of software tools, libraries, and computing power, which is why the proposed models were created, trained, and tested successfully. The major programming language was python because it is suitable with both deep learning and machine learning. Some of the most important libraries were NumPy, Pandas, SciPy, Scikit-learn, Keras, Matplotlib, and Seaborn. They helped with preparing the data, building the model, and showing it off.

The model was developed and tested using Google Colab and Jupyter Notebook. We can code and look at the results in an interactive fashion with these programs. They said that the gear should be a system with an Intel Core i5 processor, 16 GB of RAM, and a GPU-equipped framework to speed up deep learning development, cut down on computation time, and handle huge volumes of data.

4. Results and Discussion

This section shows how well the Machine Learning models and the new hybrid deep learning model H-CBLSTMNet did. Metrics are used to judge how well each model works in classifying faults by looking at the results.

1) Accuracy

Accuracy tests how right the model is overall by finding the percentage of true predictions given bearing conditions out of all predictions. In this study, accuracy is crucial for evaluating the reliability of the predictive maintenance model in differentiating between normal and problematic situations of rotating machinery. A high accuracy value suggests that the classifier performs well for all classes, which lowers the risk of misdiagnosis. The accuracy is highly crucial when it comes to industrial IoT solutions as it can be used to identify issues immediately, which will conserve resources in terms of expenses and downtime of the machines..

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

2) Loss

Loss tells us how inaccurate the guesses of the model are as we train and test the model. This demonstrates the capability of the model to learn using the information concerning vibrations.. When the loss values are smaller, the model functions better and the predictions are more accurate. In this study, loss aids in the observation of convergence and the identification of overfitting, hence guaranteeing the model's proficient generalization to unobserved machine configurations.

$$Loss = -\frac{1}{m}\sum_{i=1}^{m} \mathcal{Y}i.\log(\mathcal{Y}i) \quad (2)$$

3) Precision

Precision examines to see how many of the samples that were judged to be bad were actually bad. It is a good way to see how well the model can discover problems and how few false alarms it can send. In predictive maintenance, a high level of accuracy means that the system doesn't often send out erroneous error signals. This means that there is no need to monitor or turn off machinery for no cause. In this investigation, high precision means that the model can always find real bearing defects. This makes it safe to use in real-world businesses, where false positives can be expensive.

$$Precision = \frac{TP}{TP+FP} \quad (3)$$

4) Recall

Recall tells us the extent to which the model is sensitive to bearing faults, which is, what is it capable of discovering all the real fault situations? High recall value implies that the model does not lose many real faults and this is extremely critical as far as detecting faults in machines that move at high velocity is concerned. Recall can be considered very significant in this study as it ensures that none of the faulty bearing conditions are overlooked. This will enable businesses to not have to run into a situation where things start to go off track. Increased recall implies that the faults are detected quicker, and using the equipment is safer and more reliable.

.

$$Recall = \frac{TP}{TP+FN} \quad (4)$$

5) F1 Score

The F1-Score is a useful method of measuring Precision and Recall when the classes are not uniformly distributed. It combines both of the measurements in a single score, which ensures that false positives and false negativity are not the most significant items to examine when gauging performance. F1-Score is particularly significant to the present research as it informs

International Journal of Research and Technology (IJRT)
International Open-Access, Peer-Reviewed, Refereed, Online Journal
ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529
| An ISO 9001:2015 Certified Journal |

us about the level of faults detection by the model on the various types of bearings. The optimal approach to implement predictive maintenance is to achieve a large F1-Score, which implies that the model will always identify the problems accordingly and not commit numerous errors..

$$F1 - score = \frac{2}{\frac{1}{precision}+\frac{1}{recall}} \quad (5)$$

Table 4.1: Performance of Machine Learning Models for Bearing Fault Classification

Model	Accuracy	Precision	Recall	F1 Score
Random Forest	0.9831	0.9831	0.9831	0.9830
K-Nearest Neighbors (KNN)	0.9430	0.9422	0.9430	0.9424

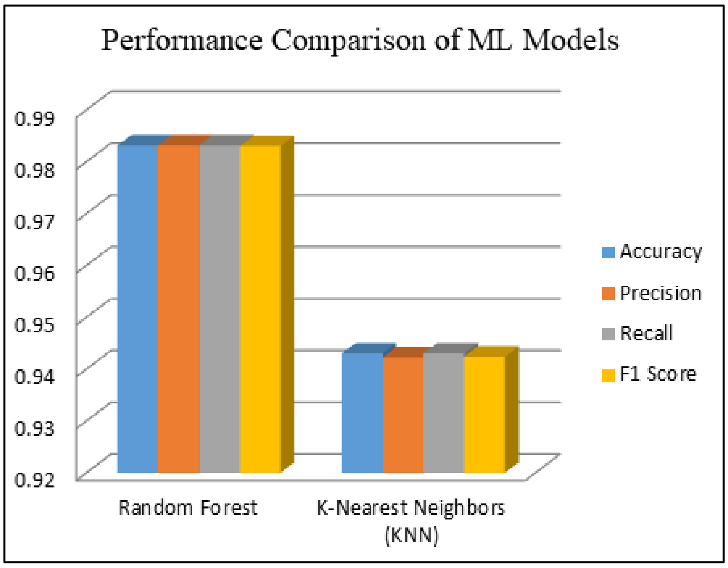

Figure 7 Performance of Machine Learning Models

Table 4.1 demonstrates the effectiveness of two machine learning tools, Random Forest and KNN in discerning the distinction between various kinds of bearing faults. The Random Forest model performed at a higher pace than all the models on all tests having the accuracy of 98.31. This demonstrates that it is able to generalize and combine features since it is built on an ensemble. Precision, Recall and F1-score values remained the same with the values being 0.9831, 0.9831, and 0.9830 respectively. This implies that the predictions were highly reliable and accurate, and with minimal errors. The KNN model, however, achieved 94.30 percent accuracy, which implies that it did not perform as well.. It also had a lower precision and F1-score. This is mostly because it uses a distance-based categorization method that is more sensitive to noise and data with a lot of dimensions. Random Forest was usually better than KNN, hence it is the best way to use ML to discover problems. But both models did well, which offers us a good starting point for comparing the suggested deep learning model's performance.

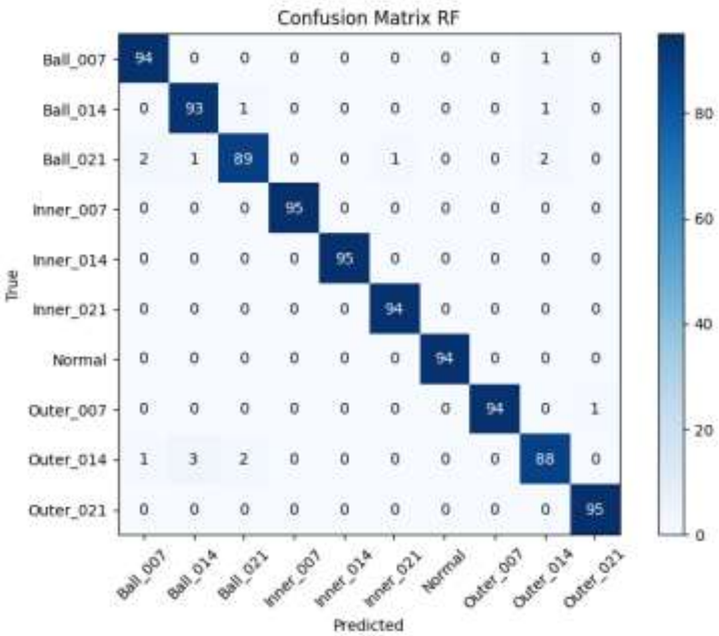

Figure 8 Confusion Matrix RF

Figure 7 shows the Random Forest model's confusion matrix. It shows how successfully the model sorts samples by showing how many were properly and mistakenly predicted in each class. The model is quite good at predicting things since it has a lot of true positives and true negatives. It makes relatively few mistakes. The model's low amount of false positives and false negatives shows that it is good at discovering patterns and class labels, which means it works well for the study's categorization goal.

International Journal of Research and Technology (IJRT)
International Open-Access, Peer-Reviewed, Refereed, Online Journal
ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529
| An ISO 9001:2015 Certified Journal |

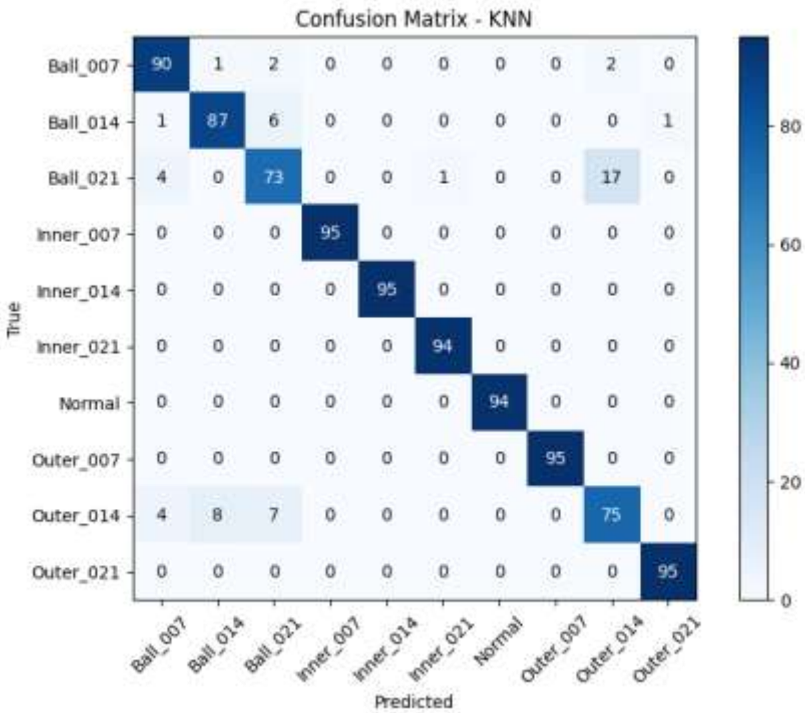

Figure 9 Confusion Matrix KNN

Figure 8 illustrates the K-Nearest Neighbors model's confusion matrix, showing how well it sorts each group. The model works well most of the time, although it produces more mistakes than the Random Forest model, indicating that it is sensitive to class borders. The model is doing fine because the number of correct and inaccurate predictions is about the same. This shows that KNN works, albeit not as well as the RF model in this investigation.

Table 4.2 Performance of Proposed H-CBLSTMNet Model on Training, Validation, and Test Data

Dataset	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Loss
Training	97.80	97.80	97.80	97.80	0.0783
Validation	96.83	96.88	96.88	96.87	0.1185
Test	98.73	98.74	98.73	98.74	0.0436

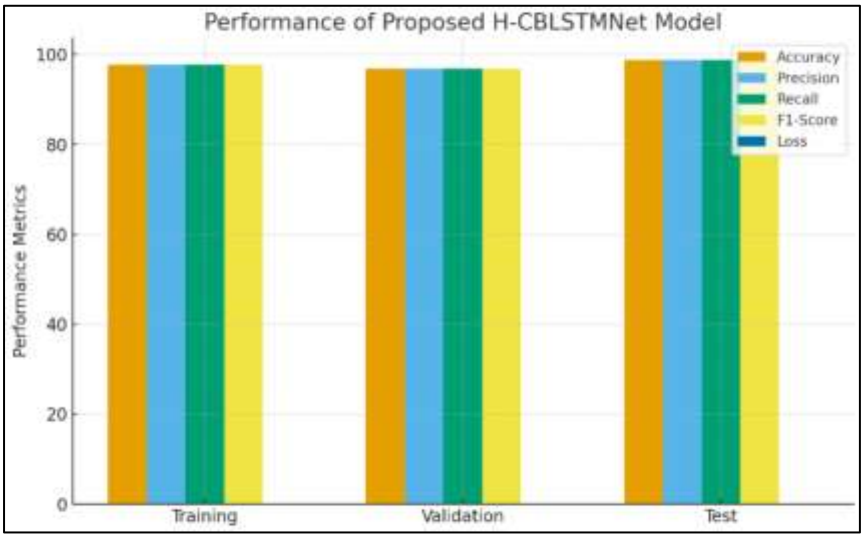

Figure 10 Performance of Proposed H-CBLSTMNet Model

Table 4.2 shows how well the proposed H-CBLSTMNet hybrid deep learning model works on training, validation, and test datasets. The model has a high classification ability, with a test accuracy of 98.73%, which is the highest of all the models tested. This shows that it is very good at finding bearing defects and may be used in a lot of different situations. The Precision, Recall, and F1-Score for the test set stayed high at 98.74%, 98.73%, and 98.74%, which means that the performance was balanced with very few false positives and false negatives. The training and validation results were also good, with 97.80% and 96.83% accuracy, respectively. This indicates that the model was not overfitted.. The model's ability to train well and make accurate predictions is further shown by the low test loss value of 0.0436. H-CBLSTMNet was much better than regular ML models since it combined CNN-based extracting features with BiLSTM-driven temporal modeling. This makes it a great model for finding smart faults.

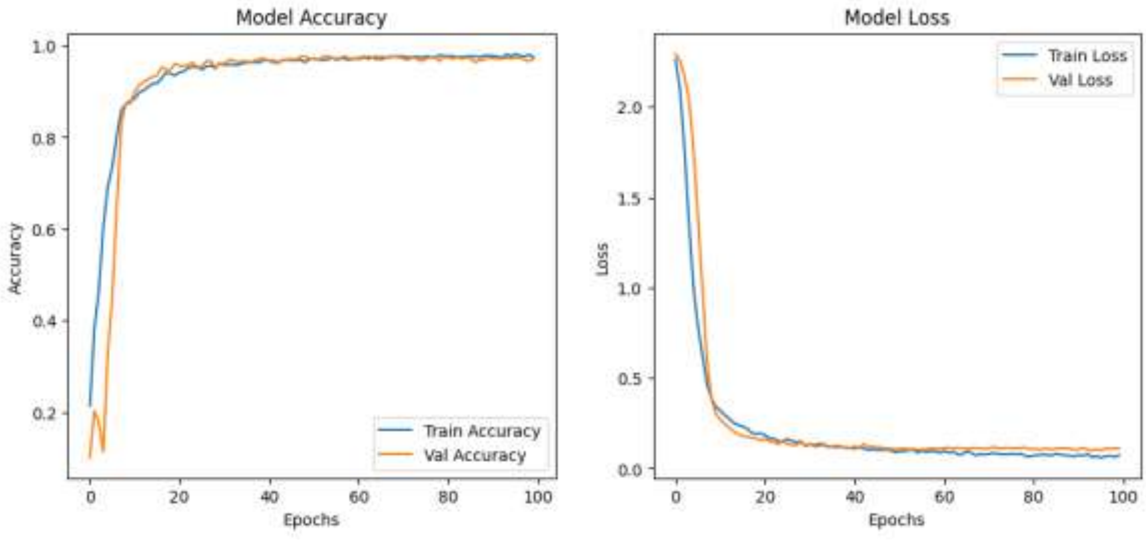

Figure 11 Accuracy and Loss graph of Proposed H-CBLSTMNet model

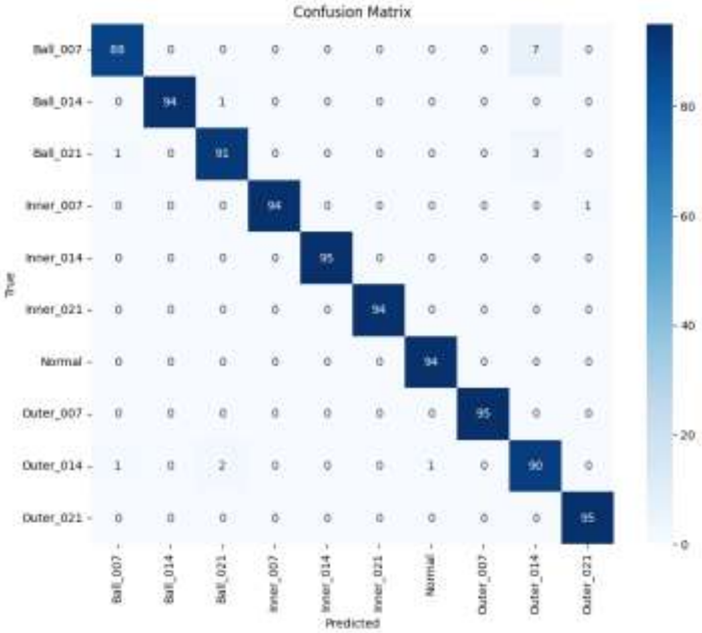

Figure 12 Confusion Matrix of Proposed Model

Figure 10 shows the accuracy and loss performance of the proposed H-CBLSTMNet algorithm on the training, validation, and test datasets. It is obvious that the model is learning steadily and not overfitting. The model gets the greatest test accuracy of 98.73%, which is backed up by consistently strong values. This means that it can classify things evenly with very few mistakes. Figure 11 shows the suggested model's confusion matrix, which shows that it can correctly categorize bearing fault categories with extremely few false predictions. The model's low test loss of 0.0436 shows that it is strong, can generalize well, and is better at diagnosing problems than typical machine learning models.

5. Conclusion

This study effectively showcased an innovative and intelligent fault diagnosis framework for bearing health monitoring utilizing the proposed Hybrid Convolutional Bidirectional LSTM Network (H-CBLSTMNet). The model manages to integrate CNN-based feature extraction with BiLSTM-based temporal learning, which enables it to directly learn both spatial as well as sequential fault features directly out of vibration data. The hybrid architecture was better than the conventional machine learning and independent deep learning models in that it was able to recognize more deep signal patterns, short-term variations and long-term temporal patterns that are important in identifying defects early. The model has performed really well with 98.73% of the tests being correct and 98.74% Precision, Recall and F1-Score being correct. These statistics indicate that the classification system is effective, and there are not so many false alarms and issues that are not detected. Generalization was also well done as the training and validation accuracy stood at 97.80 and 96.83, respectively. This proved that it was not stable and fit. A very low test loss of 0.0436 again demonstrated that the model was able to learn and remain strong. The confusion matrix proposed by the H-CBLSTMNet properly and easily classified the classes by the fault category thus indicating that H-

CBLSTMNet is applicable in the diagnostics. Conventional machine learning approaches such as the Random Forest, the SVM and KNN were however not as precise and did not extract deep discriminatory features of raw vibration signals. The reason why H-CBLSTMNet is successful is that it employs CNN layers to extract features in a hierarchical manner, BiLSTM to maintain memory on both directions, and Adam, early halting, and Dropout regularization to prevent overfitting. On the whole, the findings indicate that H-CBLSTMNet is an appropriate method of finding bearing issues in real-time and can be applied at a large scale. Since feature engineering can be automated through the model, it can be done by less number of people. This is why it is excellent with Industry 4.0 predictive maintenance systems. This helps people get ready, makes machines work better, and cuts down on downtime.

References

[1] S. Kontos, A. Bousdekis, K. Lepenioti, and G. Mentzas, “Degradation Modelling and Prognostics of Rotating Equipment with Automated Machine Learning,” *Procedia Comput. Sci.*, vol. 253, no. 2024, pp. 1640–1648, 2025, doi: 10.1016/j.procs.2025.01.226.

[2] I. Ul Hassan, K. Panduru, and J. Walsh, “Predictive Maintenance in Industry 4.0: A Review of Data Processing Methods,” *Procedia Comput. Sci.*, vol. 257, pp. 896–903, 2025, doi: 10.1016/j.procs.2025.03.115.

[3] K. M. A. Alghtus, A. Gannan, K. M. Alhajri, A. L. A. Al Jubouri, and H. A. I. Al-Janahi, “Short-Horizon Predictive Maintenance of Industrial Pumps Using Time-Series Features and Machine Learning,” pp. 1–15, 2025.

[4] A. K. Ovacıklı, M. Yagcioglu, S. Demircioglu, T. Kocatekin, and S. Birtane, “Supervised Learning-Based Fault Classification in Industrial Rotating Equipment Using Multi-Sensor Data,” *Appl. Sci.*, vol. 15, no. 13, 2025, doi: 10.3390/app15137580.

[5] S. D. Brito, G. J. Azinheira, J. F. Semião, N. M. Sousa, and S. P. Litrán, “Non-Intrusive Low-Cost IoT-Based Hardware System for Sustainable Predictive Maintenance of Industrial Pump Systems,” *Electron.*, vol. 14, no. 14, pp. 1–34, 2025, doi: 10.3390/electronics14142913.

[6] V. I. Vlachou *et al.*, “Intelligent Fault Diagnosis of Ball Bearing Induction Motors for Predictive Maintenance Industrial Applications,” pp. 1–30, 2025.

[7] J. Feng and J. Kan, “A Novel Multi-Objective Fuzzy Deep Learning Framework for Predictive Maintenance in Industrial Internet of Things,” *IEEE Access*, vol. 13, no. February, pp. 41955–41973, 2025, doi: 10.1109/ACCESS.2025.3547863.

[8] T. M. Le, H. M. Tran, K. Wang, H. V. Pham, and S. V. T. Dao, “An Internet-of-Things-Integrated Deep Learning Model for Fault Diagnosis in Industrial Rotating Machines,” *IEEE Access*, vol. 13, no. April, pp. 57266–57286, 2025, doi: 10.1109/ACCESS.2025.3553155.

[9] J. Garcia, L. Rios-Colque, A. Peña, and L. Rojas, “Condition Monitoring and Predictive Maintenance in Industrial Equipment: An NLP-Assisted Review of Signal

Processing, Hybrid Models, and Implementation Challenges," *Appl. Sci.*, vol. 15, no. 10, pp. 1–35, 2025, doi: 10.3390/app15105465.

[10] A. Moccardi, C. Conte, R. Chandra Ghosh, and F. Moscato, "A Robust Conformal Framework for IoT-Based Predictive Maintenance," *Futur. Internet*, vol. 17, no. 6, pp. 1–27, 2025, doi: 10.3390/fi17060244.

[11] C. Tsallis, P. Papageorgas, D. Piromalis, and R. A. Munteanu, "Application-Wise Review of Machine Learning-Based Predictive Maintenance: Trends, Challenges, and Future Directions," *Appl. Sci.*, vol. 15, no. 9, pp. 1–25, 2025, doi: 10.3390/app15094898.

[12] A. Lycksam, M. O'Nils, and F. Z. Qureshi, "A prognostic framework for rotating machines considering multi-component fault scenarios," *IEEE Access*, vol. 13, no. April, pp. 91682–91692, 2025, doi: 10.1109/ACCESS.2025.3572582.

[13] A. T. Abdullah, R. S. Sabeeh, H. M. Hussein, and M. J. Hussien, "A Comprehensive Review of Machine Learning Algorithms for Fault Diagnosis and Prediction in Rotating Machinery," vol. 3, no. 4, pp. 110–127, 2025.

[14] M. Nsor, "Predictive Maintenance Using Machine Learning for Engineering Systems Through Real-Time Sensor Data and Anomaly Detection Models," *Int. J. Res. Publ. Rev.*, vol. 6, no. 7, pp. 5167–5183, 2025, doi: 10.55248/gengpi.6.0725.2541.

[15] C. Eang and S. Lee, "Predictive Maintenance and Fault Detection for Motor Drive Control Systems in Industrial Robots Using CNN-RNN-Based Observers," *Sensors*, vol. 25, no. 1, 2025, doi: 10.3390/s25010025.

[16] R. Aragonés, J. Oliver, and C. Ferrer, "Enhanced Heat-Powered Batteryless IIoT Architecture with NB-IoT for Predictive Maintenance in the Oil and Gas Industry," *Sensors*, vol. 25, no. 8, 2025, doi: 10.3390/s25082590.

[17] S. K. Shil, "AI DRIVEN PREDICTIVE MAINTENANCE IN PETROLEUM AND POWER SYSTEMS USING RANDOM FOREST REGRESSION MODEL," vol. 04, no. 01, pp. 363–391, 2025, doi: 10.63125/477x5t65.

[18] M. M. R. Shamim and R. A. Ruddro, "Smart Diagnostics in Industrial Maintenance: a Systematic Review of Ai-Enabled Predictive Maintenance Tools and Condition Monitoring Techniques," *ASRC Procedia Glob. Perspect. Sci. Scholarsh.*, vol. 01, no. 01, pp. 63–80, 2025, doi: 10.63125/b4tn2x46.

[19] N. Nayak *et al.*, "Enhancing fault detection and predictive maintenance of rotating machinery with Fiber Bragg Grating sensor and machine learning techniques," *Int. J. Inf. Technol.*, vol. 17, no. 2, pp. 1225–1234, 2025, doi: 10.1007/s41870-024-02256-4.

[20] S. O. Alhuqay, A. T. Alenazi, H. A. Alabduljabbar, and M. A. Haq, "Improving Predictive Maintenance in Industrial Environments via IIoT and Machine Learning," *Int. J. Adv. Comput. Sci. Appl.*, vol. 15, no. 4, pp. 627–636, 2024, doi: 10.14569/IJACSA.2024.0150464.

[21] A. Emmanuel, "Artificial Intelligence in Predictive Maintenance for Industry," *Newport Int. J. Sci. Exp. Sci.*, vol. 6, no. 3, pp. 7–13, 2025, doi: 10.59298/nijses/2025/63.713.

[22] R. Ūselis, A. Serackis, and R. Pomarnacki, “Signal Processing Optimization in Resource-Limited IoT for Fault Prediction in Rotating Machinery,” *Electron.*, vol. 14, no. 18, 2025, doi: 10.3390/electronics14183670.

[23] R. W. Abdalah, O. F. Abdulateef, and A. H. Hamad, “A Predictive Maintenance System Based on Industrial Internet of Things for Multimachine Multiclass Using Deep Neural Network,” *J. Eur. des Syst. Autom.*, vol. 58, no. 2, pp. 373–381, 2025, doi: 10.18280/jesa.580218.

[24] *et al.*, “Predictive Maintenance in Industrial Automation: a Systematic Review of Iot Sensor Technologies and Ai Algorithms,” *Am. J. Interdiscip. Stud.*, vol. 5, no. 1, pp. 01–30, 2024, doi: 10.63125/hd2ac988.

[25] S. Hanifi, B. Alkali, G. Lindsay, M. Waters, and D. McGlinchey, “Advancements in predictive maintenance modelling for industrial electrical motors: Integrating machine learning and sensor technologies,” *Meas. Sensors*, vol. 38, 2025, doi: 10.1016/j.measen.2024.101473.

[26] T. T. Luu and D. A. Huynh, “A ResNet-based deep reinforcement learning framework using soft actor-critic for remaining useful life prediction of rolling bearings,” *Results Eng.*, vol. 27, no. August, p. 106739, 2025, doi: 10.1016/j.rineng.2025.106739.

[27] M. R and R. R. Mutra, “Fault classification in rotor-bearing system using advanced signal processing and machine learning techniques,” *Results Eng.*, vol. 25, no. December 2024, p. 103892, 2025, doi: 10.1016/j.rineng.2024.103892.

[28] Ovie Vincent Erhueh, Chukwuebuka Nwakile, Oluwaseyi Ayotunde Akano, Adeoye Taofik Aderamo, and Enobong Hanson, “Advanced maintenance strategies for energy infrastructure: Lessons for optimizing rotating machinery,” *Glob. J. Res. Sci. Technol.*, vol. 2, no. 2, pp. 065–093, 2024, doi: 10.58175/gjrst.2024.2.2.0073.

[29] L. Magadan, J. Roldan-Gomez, J. C. Granda, and F. J. Suarez, “Early Fault Classification in Rotating Machinery With Limited Data Using TabPFN,” *IEEE Sens. J.*, vol. 23, no. 24, pp. 30960–30970, 2023, doi: 10.1109/JSEN.2023.3331100.