Bearing Fault Detection based on Internet of Things using Convolutional Neural Network

Summary:

In the modern world, Industry and Machinery is very essential for our human civilization. Bearing is are industrial equipment.

Bearing Fault Detection:

Abstract(Introduction):

A bearing is a mechanical component that helps reduce friction between two surfaces that are in contact with each other, allowing for smooth movement and rotation. It typically consists of two parts – an inner race and an outer race – and a set of rolling elements or balls that move between these races. Bearings are commonly used in machines and equipment for various applications, including automotive, aerospace, industrial, and household appliances, among others. They come in different shapes, sizes, and materials, such as steel, ceramic, and plastic, depending on the specific requirements of the application. Bearings are designed to withstand high loads, temperatures, and speeds and require regular maintenance to prevent failure and prolong their lifespan.

The proposed system is efficient and effective in detecting faults in industrial machines. It is a great solution for industries where machines are too big for human monitoring. By using accelerometers sensors, the system collects data efficiently and processes it using DWT, making it easier to detect faults accurately. The study shows that custom CNN architectures perform better than transfer learning-based models, and the Convolutional 1D and 2D architectures are the best performers. This system's successful implementation can lead to reduced maintenance costs for industries and higher productivity, resulting in better outcomes for human civilization**.**

The study focused on transferring knowledge from a large dataset to a smaller dataset of industrial machinery vibration signals for fault diagnosis. The results showed that the MobileNetV2 architecture had the best performance with an accuracy of 96.88% and F1 score of 0.97 for the detection of machine faults [7]. In conclusion, the use of advanced machine learning techniques such as transfer learning and deep learning is essential for early diagnosis of faults in industrial machinery. These techniques can lead to improved productivity and a safe working environment by detecting and preventing equipment failures.

The research has shown that Artificial Intelligence-based fault detection systems, particularly those utilizing deep learning algorithms, have shown significant promise in accurately detecting faults in industrial machinery. Feature extraction from raw signals plays a crucial role in detecting faults, and transfer learning models have been shown to be effective in cases where training data is limited. The authors of this study have applied CNN algorithms to a Dataset obtained from Case Western Reserve University to detect bearing faults in machinery using a wireless IOT framework. The results of their study can help inform the development of effective fault detection systems for industrial machinery.

Data Collection and Pre-Processing

This technical description outlines a ZigBee-based wireless sensor network (WSN) designed for fault diagnosis in industrial bearings. The system utilizes accelerometer sensors to collect data, which is then transmitted via the TMS320F28335 digital signal microcontroller to a coordinator gateway using XBee radio transmitters. The network is configured using ZigBee network protocols to create a mesh network of XBee devices. The fault diagnosis model involves several steps, including applying DWT to decomposed signals, preprocessing data, splitting the dataset into testing and training, and classifying data for fault detection. The performance of the system is then evaluated. This WSN concept has the potential to improve fault diagnosis in industrial settings, increasing efficiency and reducing downtime.

Convolution 1D and Convolution 2D

The Convolutional Neural Network (CNN) is a type of neural network that is commonly used in image processing and computer vision tasks. In a CNN, the convolutional layers are the key components that perform the feature extraction process.

The Conv2D and Conv1D layers are two types of convolutional layers in CNNs. The 2D in Conv2D refers to the fact that each channel in the input and filter is two-dimensional, while the 1D in Conv1D refers to the fact that each channel in the input and filter is one-dimensional. These layers are designed to capture the spatial and temporal features respectively, in the input data.

In order to improve the performance of our Convolution 1D and Convolution 2D models, we initially normalized the data. The first hidden layer was set up with 100 nodes and implemented the RELU activation feature. MaxPooling1D was used with a pool size of 2 to minimize the dimension feature. The second and third convolution layers have 32 and 10 nodes, respectively. The signals were translated into NumPy arrays to speed up the computation.

For backpropagation, the learning rate was set at 0.001 and we used Categorical Cross-Entropy and the Adam optimizer equation to calculate the loss function. The categorical Cross-Entropy loss function is used for the multiclass classification of the dataset. After applying all of the optimizers (Adam, Nadam, Adagrad, RMSProp, Adadelta, SGD, Adamax), Adam optimizer was chosen for the highest accuracy on the dataset.

The batch size for instruction was set at 32, and the decay was set to 0.1. Tables I and II depict the convolution 2D and convolution 1D models, respectively. These tables provide a detailed overview of the architecture and hyperparameters used in each model, including the number of layers, nodes, and activation functions. By experimenting with different architectures and hyperparameters, we can optimize the performance of our models to achieve the best possible accuracy for our specific task.

MobileNetV2

MobileNetV2 has been widely adopted in the computer vision community due to its impressive balance between accuracy and efficiency, making it an ideal choice for deployment on mobile and embedded devices. Its use of depth-wise separable convolutions significantly reduces the computational cost and memory footprint of the network, while the addition of skip connections and residual blocks helps to improve its accuracy. Additionally, the optimization for quantization makes MobileNetV2 a suitable choice for deployment on devices with limited computational resources.

Performance Measures

The equations used to calculate precision, recall, f1-score, and accuracy are:

Precision = TP / (TP + FP) (Eq. 1)

Recall = TP / (TP + FN) (Eq. 2)

F1-score = 2 * (Precision * Recall) / (Precision + Recall) (Eq. 3)

Accuracy = (TP + TN) / (TP + TN + FP + FN) (Eq. 4)

In these equations, TP represents the number of true positive predictions (i.e., the model correctly predicted a positive instance), TN represents the number of true negative predictions (i.e., the model correctly predicted a negative instance), FP represents the number of false positive predictions (i.e., the model predicted a positive instance when it was actually negative), and FN represents the number of false negative predictions (i.e., the model predicted a negative instance when it was actually positive).

Precision measures how many of the predicted positive instances are actually positive, and is a useful metric when the cost of false positives is high. Recall measures how many of the actual positive instances are correctly predicted as positive, and is a useful metric when the cost of false negatives is high. F1-score is the harmonic mean of precision and recall, and is a good metric to use when you want to balance both precision and recall. Finally, accuracy measures how many instances are correctly predicted overall, and is a useful metric when the number of positive and negative instances in the dataset is balanced

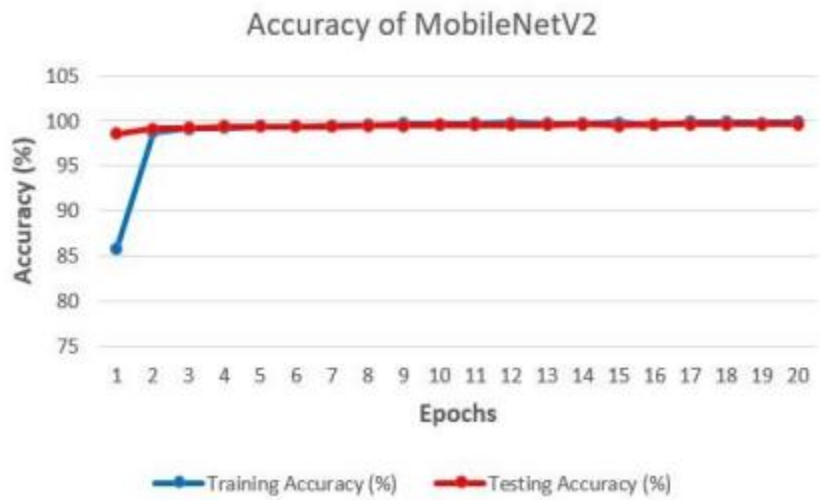
MobilenetV2 Result Analysis

The results presented in the table and figures indicate that the MobileNetV2 model achieved high accuracy and precision for all the classes in the test dataset. The highest training accuracy (99.86%) was achieved at epoch 17, and the highest validation accuracy (99.64%) was achieved at the same epoch. The precision of the model was 99.83%, indicating that the model made very few incorrect predictions.

The precision and data loss graphs in Fig. 5 show that the model's performance improved as the number of epochs increased. The data loss decreased gradually, and the accuracy increased for both the training and test sets. This suggests that the model learned from the data and was able to generalize well to unseen data.

The classification report for each class in Fig. 6 shows that the model had high performance for all the classes individually. The heatmap in Fig. 7 shows the number of samples correctly and incorrectly classified for each class. The model correctly predicted a large number of samples in each class, but there were some samples that were misclassified.

Overall, the results suggest that the MobileNetV2 model was effective in detecting faults in the bearings and had high accuracy and precision. However, it is important to note that the model may not perform as well on data from different sources or in different conditions.


Accuracy of MobileNetV2
105
100
95
90
85
80
75
Accuracy (%)
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
Epochs
Training Accuracy (%)
Testing Accuracy (%)

(a)

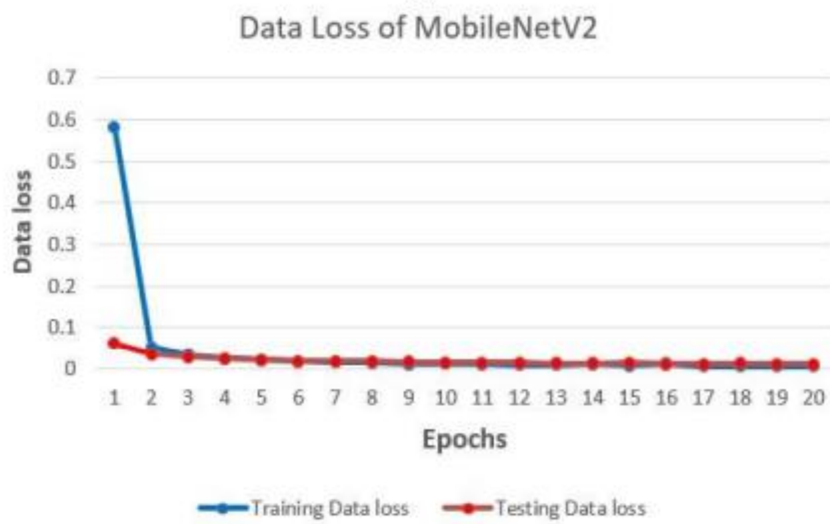

Data Loss of MobileNetV2
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Data loss
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
Epochs
Training Data loss
Testing Data loss

Fig. 5. (a) Accuracy for each Epoch, (b) Data Loss Graph, for MobileNetV2 Architecture.

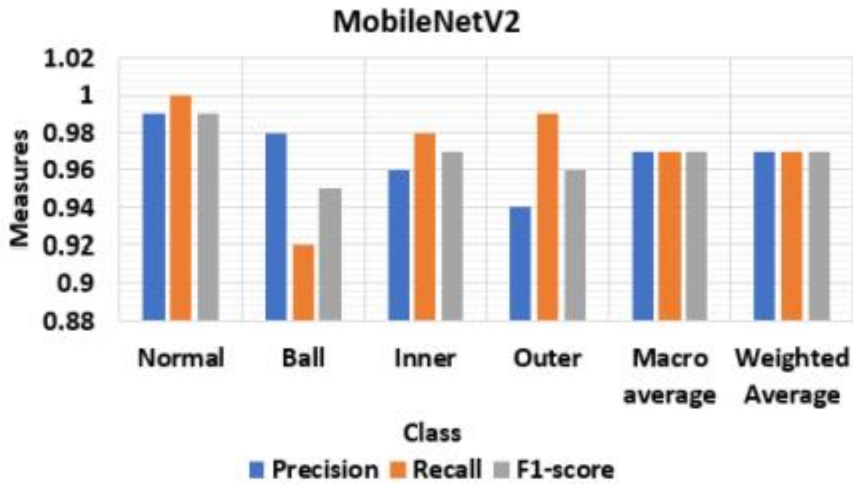


Fig. 6. Classification Report for each Class.

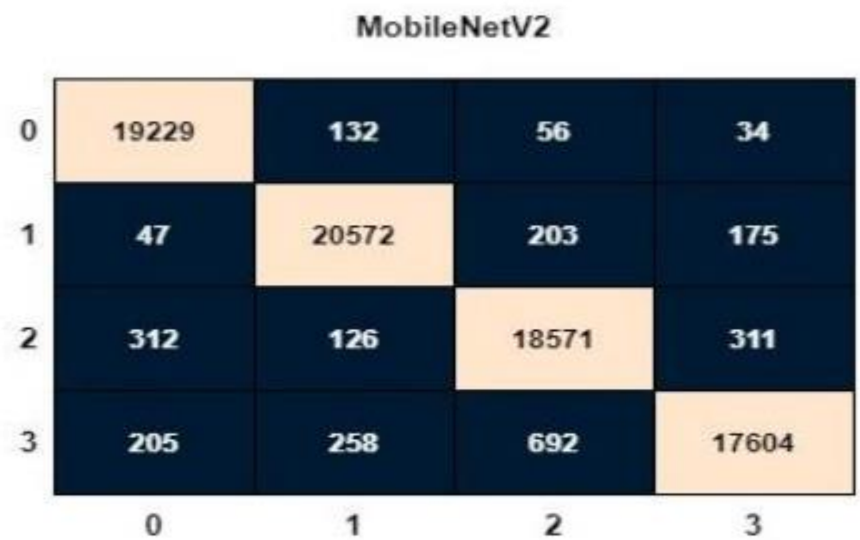


Fig. 7. Heat Map for MobileNetV2.

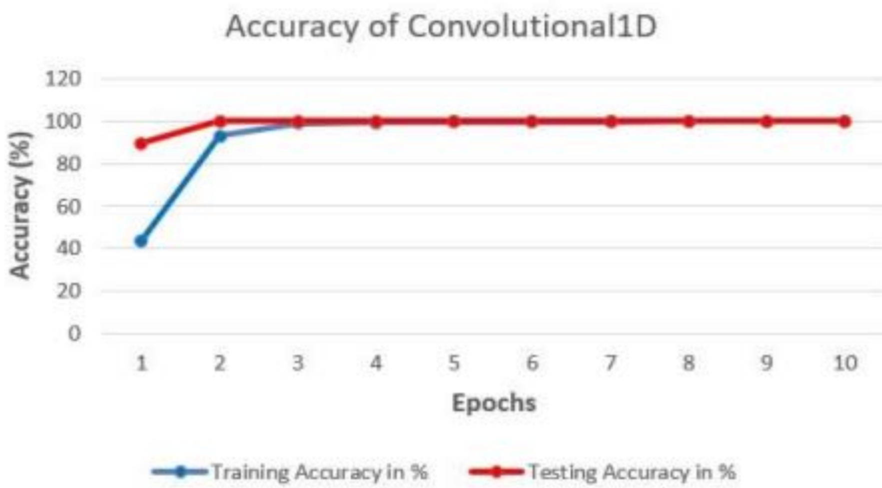
Convolution 1D result Analysis

Table IV shows the performance metrics obtained from the Convolution 1D model for 10 epochs. The model was trained on Set 3, while Sets 1 and 2 were used for testing.

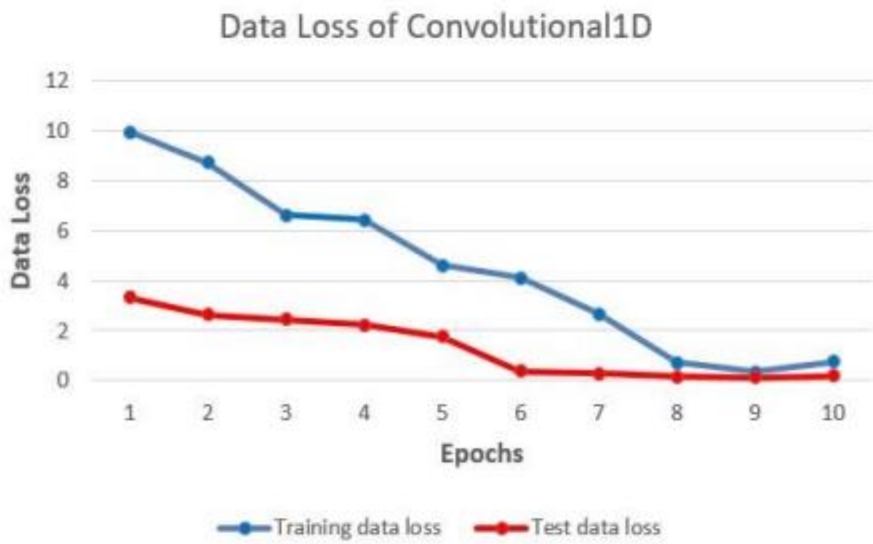
The training data loss and accuracy improved significantly over the 10 epochs, with the loss decreasing from 9.96 in the first epoch to 0.77 in the 10th epoch and the accuracy increasing from 43.88% in the first epoch to 99.97% in the 10th epoch.

For the test data, the model achieved high accuracy, with the highest accuracy of 100% obtained in the last three epochs. The data loss also decreased over the epochs, with the lowest value of 0.13 obtained in the 9th epoch.

The results suggest that the Convolution 1D model performed well on the dataset and was able to generalize well to the test data. The improvement in accuracy and decrease in data loss over the epochs suggest that the model was learning from the data and improving its predictions.

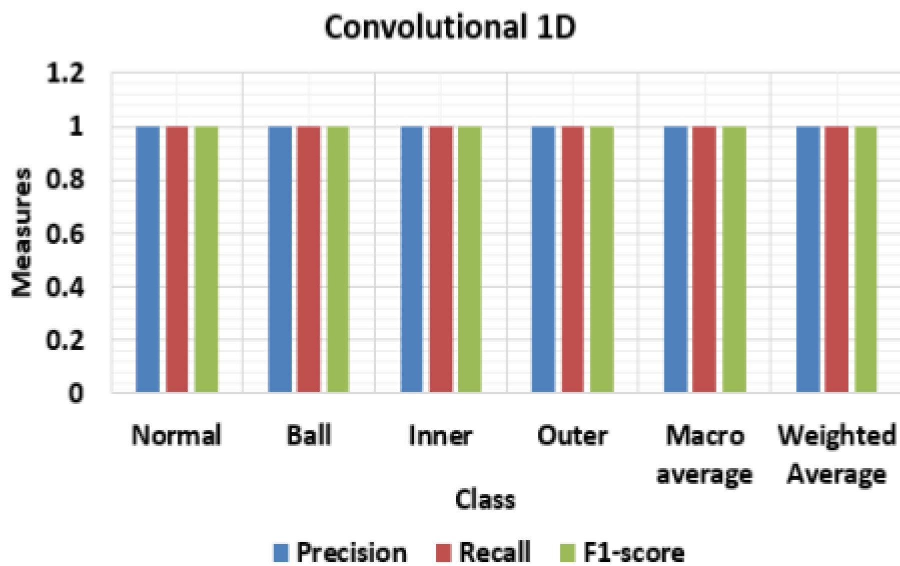


(a)



(b)

Fig. 8. (a) Data Accuracy, (b) Data Loss Graph, of Convolution 1D.



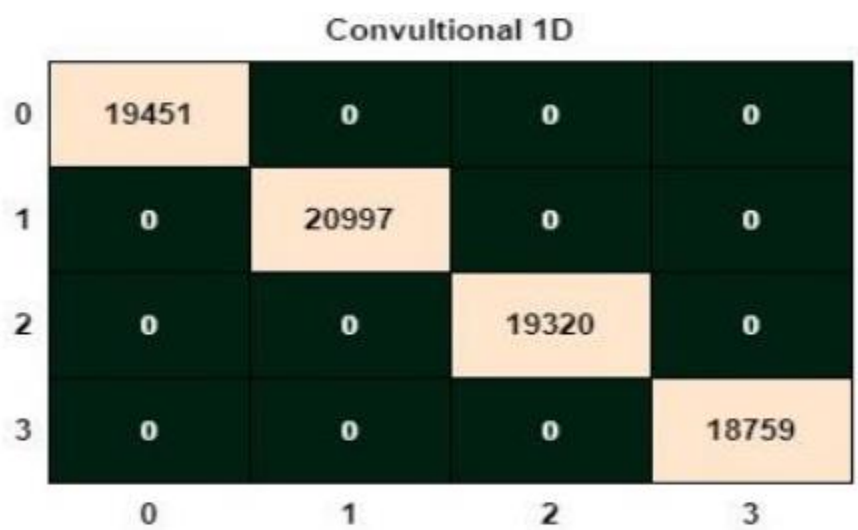
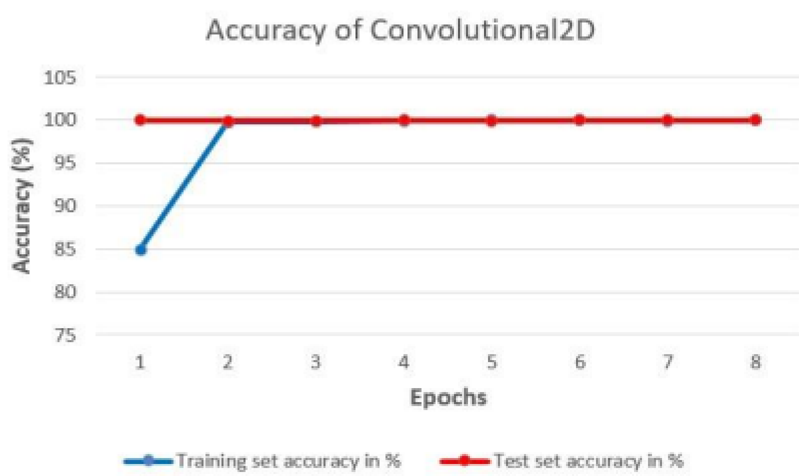


Fig. 10. Heat Map for Convolution 1D.

Convolution 2D Result Analysis

Based on the results presented in Table V, the Convolution 2D model achieved a high accuracy of 100% for both the training and testing sets. The data loss for the test set is also much lower compared to the Convolution 1D model. The data accuracy and loss graph in Fig. 11 shows that the model's accuracy rate stays consistently high for the test set, and the data loss stays consistently low on each epoch. The classification report in Fig. 12 also indicates that the Conventional 2D model shows a perfect performance score throughout all the classes of the test dataset, similar to the Conventional 1D model. Overall, the Convolution 2D model seems to perform better than the Convolution 1D model in terms of accuracy and data loss for this particular dataset.



(a)

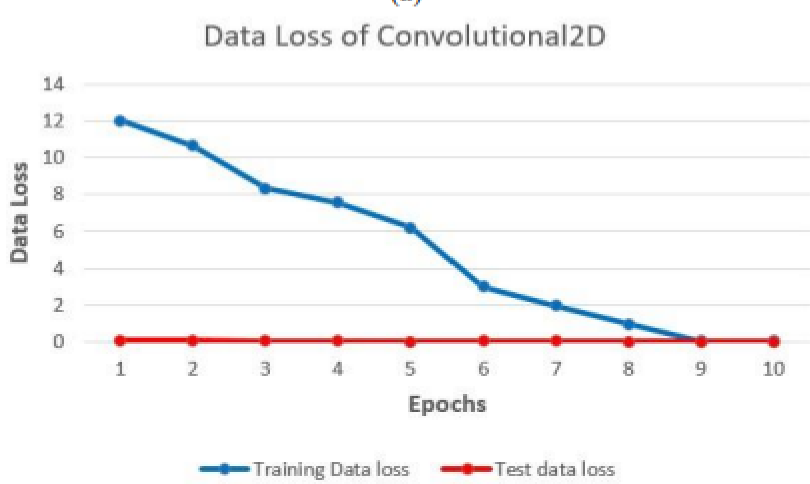


Fig. 11. (a) Data Accuracy, (b) Data Loss Graph using Convolution 2D.

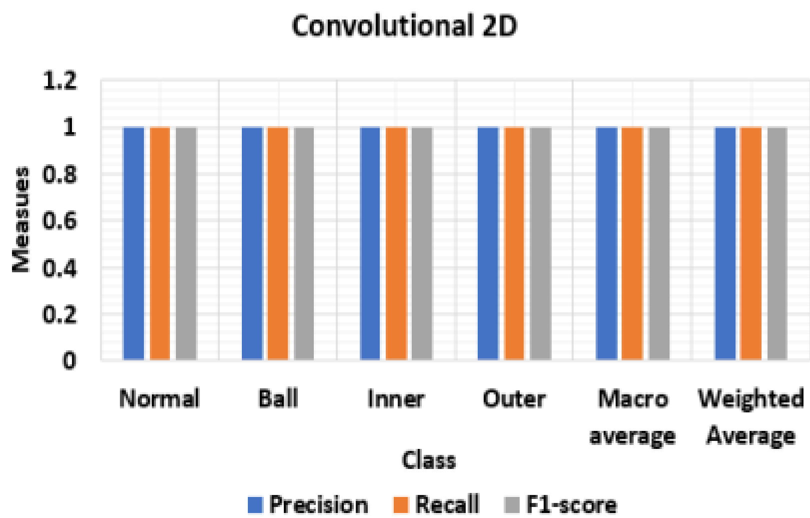


Fig. 12. Classification Report of Convolution2D.

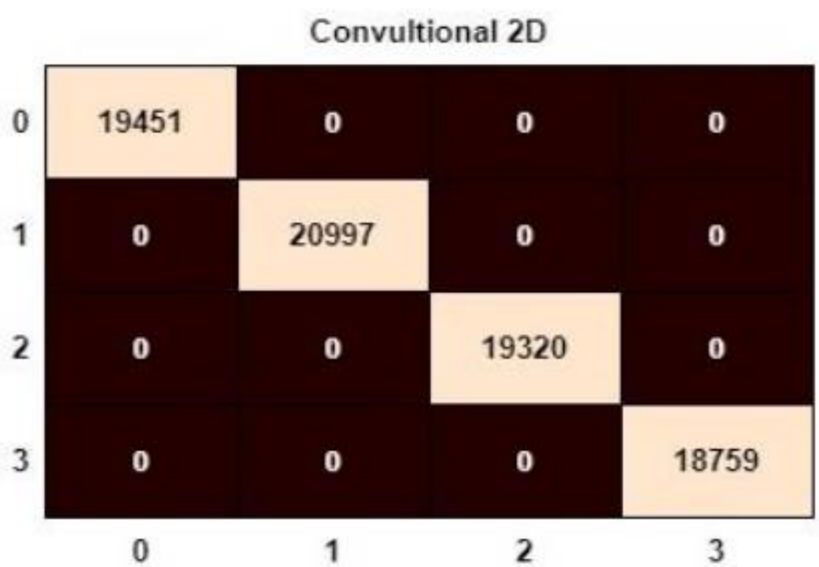


Fig. 13. Heat map for Convolution 2D.

Based on the heat map, it appears that the Convolution 2D model has performed well in predicting most of the samples correctly. However, there are some instances where it has wrongly predicted the samples. For example, in the normal class, it has wrongly predicted Nil samples, which means that the model has predicted some normal samples as abnormal. Similarly, in the ball class, some Nil samples have been wrongly predicted, indicating that the model has predicted some abnormal samples as normal.

Overall, the Convolution 2D model has shown a high level of accuracy in predicting the samples from each class. For example, it has properly predicted 100% of the samples from the inner and outer classes. However, there is still room for improvement in terms of reducing the number of wrongly predicted samples.

Result Discussion and Comparison

The table VII in the study compares the proposed system's performance with other existing systems used for fault detection in machinery. The proposed system, using MobileNetV2, Convolutional 1D, and Convolutional 2D, achieved a fault detection accuracy of 99.64% and 100% accuracy for the three algorithms, respectively. On the other hand, the existing systems, such as SVM, CNN, SVM+PCA, Stack Denoising Auto Encoder, GAN, DNN, ANN, and ANN+10-fold Cross Validation, achieved accuracy ranging from 71.67% to 98.33%. It is observed that the proposed system's performance is much better than the existing techniques, indicating that the proposed system can detect faults with the highest accuracy. The authors suggest that the proposed system can be used to make the fault detection process automatic and efficient, leading to safer industrial machinery.

Conclusion

The article describes a method for detecting bearing defects using accelerometer sensors and wavelet transformation signal processing. The system uses a ZigBee-based wireless sensor network to send data to a diagnostic server for analysis. Three different convolutional neural network (CNN) models, including the MobileNetV2 architecture and two custom 1D and 2D deep CNN models, were tested using a dataset consisting of four types of fault signals.

The results showed that the MobileNetV2 architecture achieved 99.64% accuracy and 99.83% precision in identifying bearing problems, while the custom 1D and 2D CNN models achieved up to 100% accuracy and precision. The authors suggest that the proposed architecture could be further improved by testing it with additional parameters and by implementing hybrid transfer learning models and a reliable IOT framework.

Overall, the method presented in the article provides a promising approach for detecting bearing defects and preventing equipment failure and accidents. The use of accelerometer sensors and wavelet transformation signal processing, combined with CNN models, allows for efficient and accurate monitoring of mechanical equipment.