DEGREE PROJECT

Predictive Maintenance in Industrial Machinery using Machine Learning

Jasim Aftab Abbasi

Master Programme in Data Science
2021

Luleå University of Technology
Department of Computer Science, Electrical and Space Engineering

Abstract

Background: The gearbox and machinery faults prediction are expensive both in terms of repair and loss output in production. These losses or faults may lead to complete machinery or plant breakdown.

Objective: The goal of this study was to apply advanced machine learning techniques to avoid these losses and faults and replace them with predictive maintenance. To identify and predict the faults in industrial machinery using Machine Learning (ML) and Deep Learning (DL) approaches.

Methods: Our study was based on two types of datasets which includes gearbox and rotatory machinery dataset. These datasets were analyzed to predict the faults using machine learning and deep neural network models. The performance of the model was evaluated for both the datasets with binary and multi-classification problems using the different machine learning models and their statistics.

Results: In the case of the gearbox fault dataset with a binary classification problem, we observed random forest and deep neural network models performed equally well, with the highest F1-score and AUC score of around 0.98 and with the least error rate of 7%. In addition to this, in the case of the multi-classification rotatory machinery fault prediction dataset, the random forest model outperformed the deep neural network model with an AUC score of 0.98.

Conclusions: In conclusion classification efficiency of the Machine Learning (ML) and Deep Neural Network (DNN) model were tested and evaluated. Our results show Random Forest (RF) and Deep Neural Network (DNN) models have better fault prediction ability to identify the different types of rotatory machinery and gearbox faults as compared to the decision tree and AdaBoost.

Keywords: Machine Learning, Deep Learning, Big Data, Predictive Maintenance, Rotatory Machinery Fault Prediction, Gearbox Fault Prediction, Machinery Fault Database, Internet of Things (IoT), Spectra quest machinery fault simulator, Cloud Computing, Industry 4.0

Acknowledgments

I would like to thank my supervisor Muhammad Hanif who allowed me to work under his supervision. I am grateful to you for your consistent encouragement, polite behavior, guidance, and overall dynamic supervision throughout the whole process. Without your immense experience, scientific ingenuity, and valuable advice this work would never have been possible.

I am also very thankful to my teachers during my degree program who taught me different courses and concepts related to machine learning and data analysis. I also want to thank my friends with whom I have taken several of my course projects and have had great discussions and teamwork.

I am thankful for my department and Lulea University; it was a great experience to be a student at this renounced institution. Moreover, I must acknowledge my wife and family for their support during this time.

Jasim Aftab Abbasi

Uppsala Sweden

September 2021

Contents

List of Abbreviations

AI	Artificial Intelligence
ADA	Adaptive
ANN	Artificial Neural Network
ARIMA	Auto Regressive Integrated Moving Average
AUC	Area Under Curve
BD	Big Data
CM	Confusion Matrix
CRISP-DM	Cross Industry Standard Process for Data Mining
DL	Deep Learning
DNN	Deep Neural Network
DT	Decision Tree
ERR	Error Rate
FN	False Negative
FP	False Positive
FPR	False Positive Rate
GPU	Graphics Processing Unit
HMA	Horizontal Misalignment
IoT	Internet of Things
KNN	K-Nearest Neighbours
MAFAULDA	Machine Fault Database
MFP	Machinery Fault Prediction
MFS	Machine Fault Simulator
ML	Machine Learning
MLP	Multilayer Perceptron
MSE	Mean Squared Error
NaN	Not a Number
NN	Neural Network
OBF	Overhang Bearing Fault
PCA	Principal Component Analysis
PdM	Predictive Maintenance
ReLU	Rectified Linear Unit
RF	Random Forest
RL	Reinforcement Learning
ROC	Receiver Operating Characteristic
SD	Standard Deviation
SBM	Similarity Based Models
TN	True Negative
TP	True Positive
UBF	Underhang Bearing Fault
VMA	Vertical Misalignment

1 Introduction

1.1 Motivation

The motivation for the study was to learn about the different types of industrial maintenance and their challenges. Furthermore, to learn and apply advanced analytics techniques and machine learning algorithms to predict the faults in industrial machinery. That will help the maintenance team to repair and schedule the maintenance ahead of problems to avoid any breakout in the plant or production line.

1.2 Background

Nowadays machines play a very important role in our daily lives. We rely on machines whether we can travel or fly from one place to another or construct houses, roads, or build infrastructure. Machines not only reduce time but also increase productivity.

The automotive and manufacturing industries heavily rely on different types of machines. Few of the machines used in these industries are easy to use and operate and some of them are very complex and require regular maintenance to perform their daily operations. This maintenance reduced productivity and increased the maintenance cost [1].

With the current situation of the COVID19 pandemic, most of the industries have transformed towards digitization. There is a need to automate the manual maintenance process as well. Using the advanced analytics techniques, to ensure critical asset reliability, and support on-demand manufacturing requirements [1].

Predictive maintenance was originally used for the oil and gas industry [1], but now with the internet of things (IoT) and new technologies such as cloud computing, big data tools, artificial intelligence, machine learning, industry 4.0, and sensors have brought the cost-effective predictive analytics to other new domains as well [2][3]. Automotive industries are also moving from reactive to predictive maintenance.

1.3 Problems Definition

Gearbox and rotatory machines are the most essential components in industrial machinery and play an important role in different industrial applications. Some of the applications of these components are in automotive industries, oil and gas, wind turbines, manufacturing industries, hydropower, mining, recycling plant, and so on.

The gearbox and rotatory machinery faults are expensive both in terms of repair and loss output in production. Sometimes these losses or faults may lead to complete machinery or plant breakdown. To avoid critical damage and sudden breakdown, the faults in these components should be detected as early as possible.

1. The gearbox is a binary classification problem; we can avoid the loses by predicting the health condition of the gearbox such as
    - Normal

    - Broken gearbox teeth
2. Rotatory machinery is a multiclassification problem, we can avoid the rotatory machines losses by predicting the normal operations and faulty states of machinery such as
    - Normal
    - Imbalanced
    - Horizontal misalignment
    - Vertical misalignment
    - Underhang bearing faults
    - Overhang bearing faults

1.4 Proposed Solutions

The popularity of machine learning (ML) increases rapidly in industrial automation. Now it is affordable to get the data from sensors or IoT devices and store it in a database. The availability of this historical data makes it easier to build and train the ML models and predict the current and future state of industrial machines. It helps the technical team to avoid unscheduled maintenance.

Our solution is based on ML and deep learning (DL) techniques such as decision tree, adaboost, random forest and deep neural network (DNN) to predict the different types of faults in these industrial components. This will help the maintenance team to repair or replace the components before the faults happen.

2. Related Work

There are several studies published previously on the detection of faults in gearbox and the rotatory machinery by several groups using multiple techniques, as summarized below briefly.

F. Ribeiro et al. have used non-machine learning techniques such as similarity-based models (SBM) to automatically classify the faults in rotatory machinery [4]. As a result, they classify the faults with an accuracy of 96.43 percent.

In another study by A. Alzghoul et. al, the authors classified the rotatory faults with the accuracy of 97.1 percent using Artificial Neural Network (ANN) [5]. Like our study, MAFAULDA [6] machine fault database was used in both studies [4, 5].

Similarly, signal processing-based preprocessing algorithms and neural networks has been used to classify the gearboxes faults in another study by W.J. Staszewski et.al [7]. These models detect and classify the gearbox faults without any errors.

Zhang Qiang et.al has shown to use self-organizing map-based fault models to detect the gearbox faults with an accuracy of 95 percent [8].

With these emerging techniques and methodologies, there are still several challenges such as as computing resources and programming methods as discussed in detail in one of the study by S. R. Saufi et al. in 2019 [29]. In this study, they highlighted the challenges of machinery fault detection using deep learning. The main challenges of implementing a deep learning-based system for machinery fault prediction required high performance resources such as a GPU-based system [29].

Another challenge while performing this type of studies is at its architecture level to train the DL model. Selection of activation function and training the model required prior knowledge. Now a days different types of programming tools are using while implementing this type of system. Each programming environment have different coding styles. It might affect the fault diagnostic performance of the model. To build the DL model required huge amount of historical data to train and test the system [29].

In a more recent study published in 2021 by S. Ayva, comparative analysis and evaluation of several ML algorithms was performed by Serkan Ayva et al.[30] .Their results showed that random forest (RF) outperformed all the other algorithms studied. This enabled them to incorporate the best performing machine-learning model into the production system in the factory [30].

.

3 Industrial Maintenance and Machine Learning

3.1 Maintenance

The maintenance cost in many industries is higher than operational and production costs due to premature equipment failure [9]. The profitability of any industry generally depends on the maintenance process.

Normally maintenance in industries happens when the equipment reaches a certain age or stops working [10]. It is good to do scheduled maintenance, but it doesn't provide any information about the equipment's health in the future. To optimize the production lines and equipment reliability, different types of maintenance can be performed based on the resource. The most common types of industrial maintenance are **Figure 3.1**

1. Reactive Maintenance
2. Preventive Maintenance
3. Predictive Maintenance

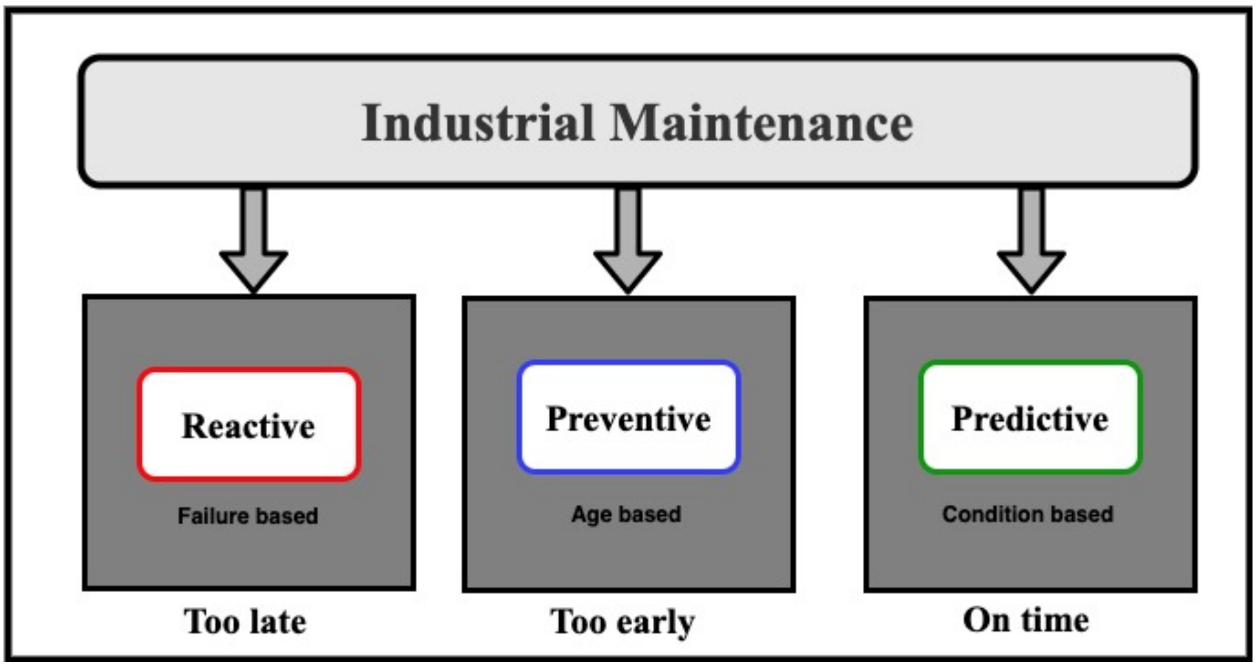

Figure 3.1: Types of industrial maintenance

3.1.1 Reactive Maintenance

In this approach, maintenance can be performed when components or machinery have a problem or stop working. Normally maintenance will perform after the equipment failure as shown in **Figure 3.2.** Although the component or machine is used full lifespan, drawbacks of this approach are

- Unscheduled maintenance
- Downtime is increased

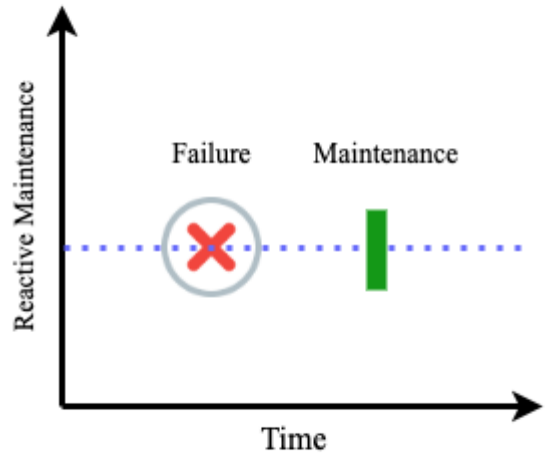

Figure 3.2: Reactive maintenance overview

3.1.2 Preventive Maintenance

In this approach, the machine or component is replaced in advance before it fails. It helps to avoid unscheduled maintenance. The maintenance will perform during the regular interval as shown in **Figure 3.3.** The drawback of this approach is [11,12,13]

- The component or machine is not fully utilized
- Over maintenance is performed

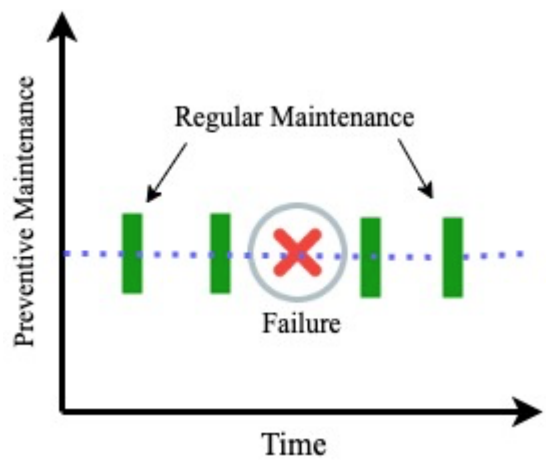

Figure 3.3: Preventive maintenance overview

The drawbacks of regular maintenance are

- Breakdown time is increased
- Productivity is reduced due to regular maintenance
- Over maintenance of some equipment or machinery
- Operation cost is an increase
- The life span of a machine is decreased
- More skilled labor is needed to maintain the equipment

3.1.3 Predictive Maintenance

It predicts the fault and performs the maintenance on the machine or equipment before the fault or failure happens as shown in the **Figure 3.4** . Only the components or machines can replace which is going to fail soon. It extends the life span of the equipment. There are several advantages of predictive maintenance [13,14,15] such as,

- It can reduce the unplanned downtime
- It can help to identify fault or equipment health by condition monitoring to avoid costly equipment failure
- It decreased the planned downtime by reducing inspection and premature repair

Predictive maintenance system is an IoT based system. The drawback of this approach is the initial cost to build such a system is very high.

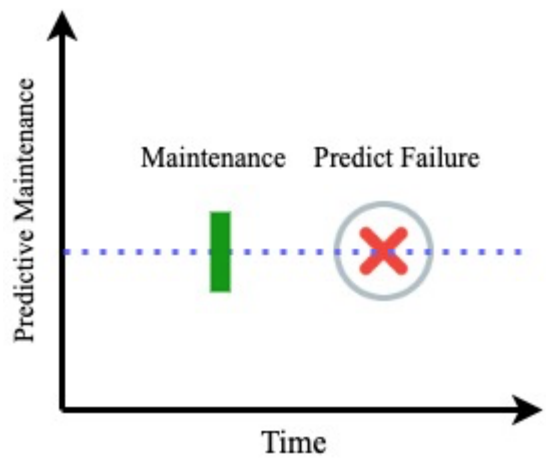

Figure 3.4: Predictive maintenance overview

3.2 Machine Learning (ML)

IoT and cloud computing make machine learning possible in manufacturing and other industries. Now it is much easier to get the data from the industrial equipment with IoT devices. These data from the industrial equipment will help us to build the ML models to predict the faults. ML transforms some of the tasks to a machine that was previously not possible with humans [16].

3.2.1 Types of Machine Learning

The ML is of three types

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning (RL)

1. **Supervised Learning**

   Supervised learning techniques are easy to understand and implement. Labeled data is provided to the ML models [17,18]. It means both training and validation data are labeled. The training datasets comprise both inputs and target outputs in supervised learning as shown in **Figure 3.5,** which allow the model to learn and improve over time. When the model is fully trained it will predict the new or unseen data with a good label. It can be used for both classification and regression problems. The algorithms in supervised learning are decision trees, random forest, support vector machine, navies byes, linear regression, logistic regression, etc.

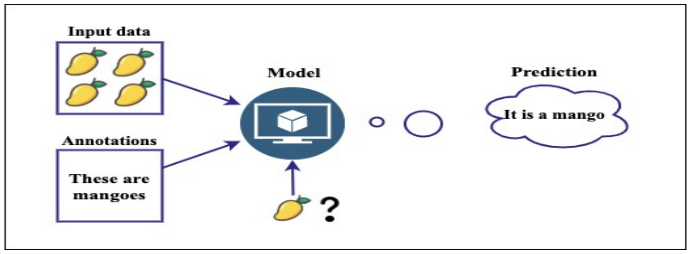

Figure 3.5: Supervised learning

2. **Unsupervised Learning**

In this approach the user does not need to provide the label data to the model, it works with unlabeled data [19]. It allows the model to detect patterns and information on its own **Figure 2.6**. It is useful to find the unknown patterns in the data. The algorithms in unsupervised learning are clustering, K- Nearest Neighbors (KNN), anomaly detection, Principal Component Analysis (PCA), etc.

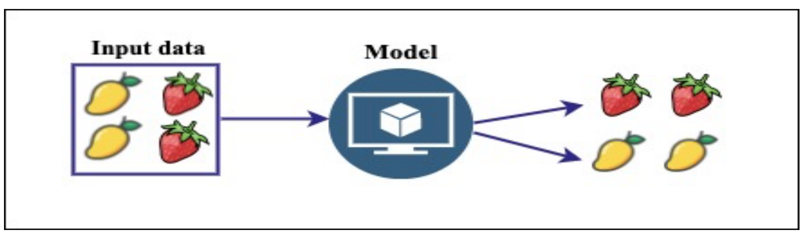

Figure 3.6: Unsupervised learning

3. **Reinforcement Learning**

RL is a type of ML and does not require a lot of training data. Instead of environments are given to the RL models, the agent learns from its environment by trial and error to achieve goals and get rewards **Figure 3.7**.

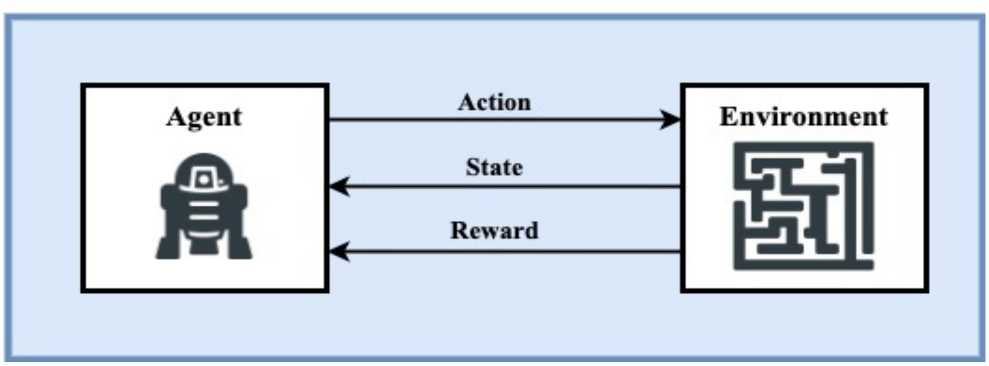

Figure 3.7: Reinforcement learning

4 Dataset and Faults

Data is the core component of any ML/DL model. Quality data is required to perform these models efficiently. The performance of the ML/DL model can improve by integrating more data into the ML/DL system. The data can be of many forms, but the ML model mainly rely on

- Numerical data
- Text data
- Categorical data
- Time series data

4.1 Experimental Setup

Spectraquest provides different types of simulators for training and studying industrial machine behaviors. These simulators accelerate learning and help to understand the different types of fault in industrial machinery [20]. The data we used to train and test the ML model was taken from these simulators

- SpectraQuest's Gearbox Fault Diagnostics Simulator
- SpectraQuest's Machinery Fault Simulator

4.2 Gearbox Dataset

The gearbox dataset used in this study is publicly available at OpenEi [21]. The data was recorded by OpenEi [21] with the four vibration sensors placed in different directions on spectra quests gearbox fault diagnostics simulator [20]. The dataset is recorded with a different load from 0 to 90 percent and contains information about the health conditions of the gearbox based on the vibrational sensors reading. Gearbox dataset describes only two states of gearbox such as

- Normal
- Broken teeth

4.3 Machinery Fault database

The data from spectraQuest Machinery Fault Simulator (MFS) are collected by sensors and stored in the machinery fault database [6]. The database contains 1951 multivariate time series data comprised of six different simulated states such as

- Normal
- Horizontal misalignment
- Vertical misalignment
- Imbalance faults
- Underhang bearing fault
- Outer bearing faults

The rotatory machinery faults database contains the following percentage of each category of data as shown in **Figure 4.1.**

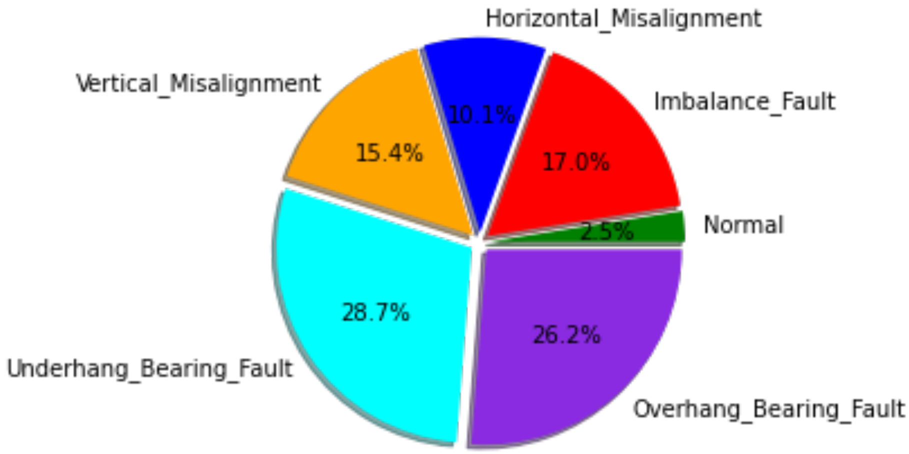

Figure 4.1: Rotatory machinery data percentage in machine fault database (MAFAULDA)

The rotatory machinery database contains the least amount of class normal data and maximum class underhang bearing faults data. The summary of the measurements is shown in **Figure 4.2**

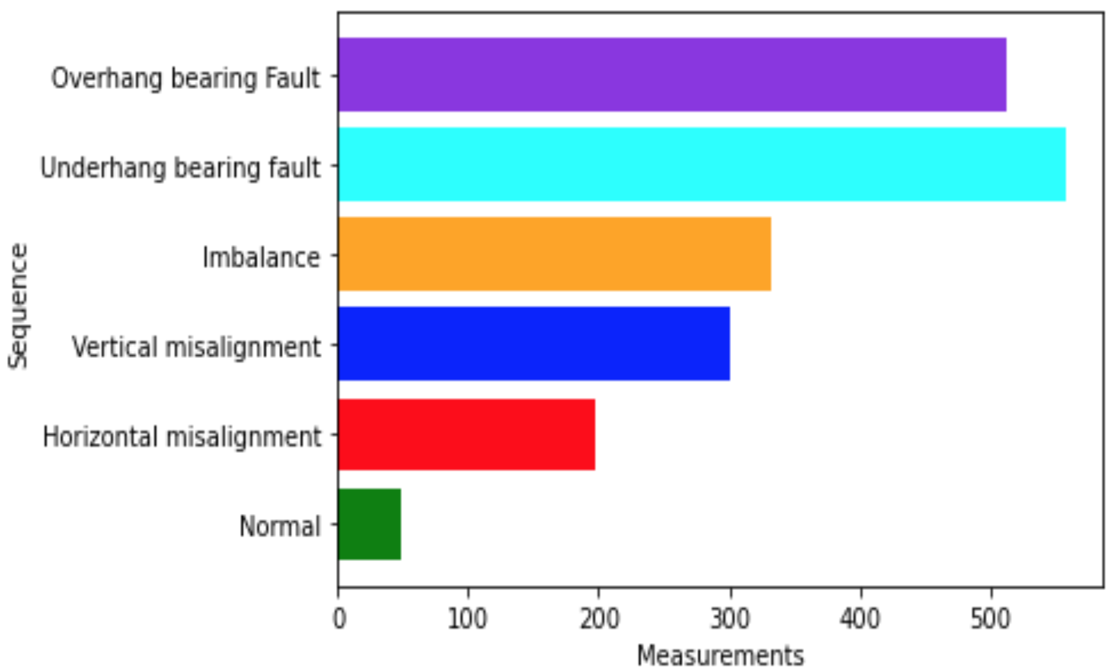

Figure 4.2: Summary of six states of rotatory machines measurement

4.4 Rotatory machine states

The data stored in the machinery fault database is acquired with the help of six accelerometers, a microphone, and a tachometer attached to the machine fault simulator [4]. It contains a total of 1951 scenarios as shown in **Figure 4.2.** The data describe the normal and five faulty states of the rotatory machine.

4.4.1 Normal

The normal sequence means without any fault. The 49 measurements of the normal sequence were used in this study as shown in **Figure 4.2**. These sequences have been recorded with fixed rotation speed (range 737-3686 rpm) [6].

4.4.2 Imbalance

The total number of imbalance faults was 333 measurements [6]. The data was recorded with the load values (6g to 35g) as shown in **Table 4.1**

Table 4.1: Summary of imbalance measurement with different load values

Weights (g)	6	10	15	20	25	30	35	**Total**
Measurements	49	48	48	49	47	47	45	333

4.4.3 Horizontal misalignment

The number of horizontal parallel misalignment was 197 which was induced by each horizontal shift by the motor shaft shifting horizontally 0.5mm,1.0mm,1.5mm, and 2.0mm into MFS **Table 4.2**.

Table 4.2: Summary of horizontal misalignment measurement with different values

Misalignment (mm)	0.50	1.00	1.50	2	**Total**
Measurements	50	49	49	49	197

4.4.4 Vertical misalignment

The number of vertical parallel misalignment was 301 which was induced by each vertical shift by the motor shaft shifting horizontally 0.51mm,0.63mm,1.27mm,1.40mm,17.8mm, and 1.90mm into MFS **Table 4.3**.

Table 4.3: Summary of vertical misalignment measurement with different values

Misalignment (mm)	0.51	0.63	1.27	1.40	1.78	1.90	**Total**
Measurements	51	50	50	50	50	50	301

4.4.5 Underhang bearing fault

In rotating machinery bearing is one of the most complex elements. Bearing faults are primarily causing failures in rotating machinery. When the bearing is placed between the rotor and motor in

MFS. The underhang bearing fault has 558 total sequences with varying weights (0g, 6g, 20g, 25g).

4.4.6 Overhang bearing fault

When the rotor is placed between the bearing and motor in MFS. The overhang-bearing fault has 513 total sequences with varying weights (0g, 6g, 20g, 25g).

5 Methods

Nowadays Artificial Intelligence (AI) has become popular in many other industries such as manufacturing and smart factories. The Internet of Things (IoT), Big Data (BD), and cloud computing make it more accessible to small industries as well. Machines in manufacturing industries have become smarter than before due to IoT, AI, and big data.

In recent times most of the manufacturing industries are transferred from preventive to predictive maintenance. This not only increases their productivity but also reduces cost. ML plays a significant role in such innovations. It also helps them to improve decision-making and accelerate discovery processes in manufacturing sectors. In the past different techniques have been used for industrial maintenance [4,5,7,8,22,23,24,25, 29,30].

In our study, we have used both classical machine learning and deep learning approaches to predict the fault in industrial machines as shown in the **Figure 5.1**. We followed the design of the Cross Industry Standard Process for Data Mining (CRISP-DM) model, which includes the following steps/processes; (i) Business understanding: which includes the understanding of the industrial maintenance and their challenges and proposed solution; (ii) Data understanding: includes information/knowledge of our datasets; (iii) Data preparation includes the preprocessing steps that helped to prepare the data for downstream analysis ;(iv) Modeling includes the steps where different analysis models and algorithm were applied; (v) Evaluation includes the step where we evaluated the performance of the different models; (vi) Deployment includes our final model that was selected and applied to the data for the solution.

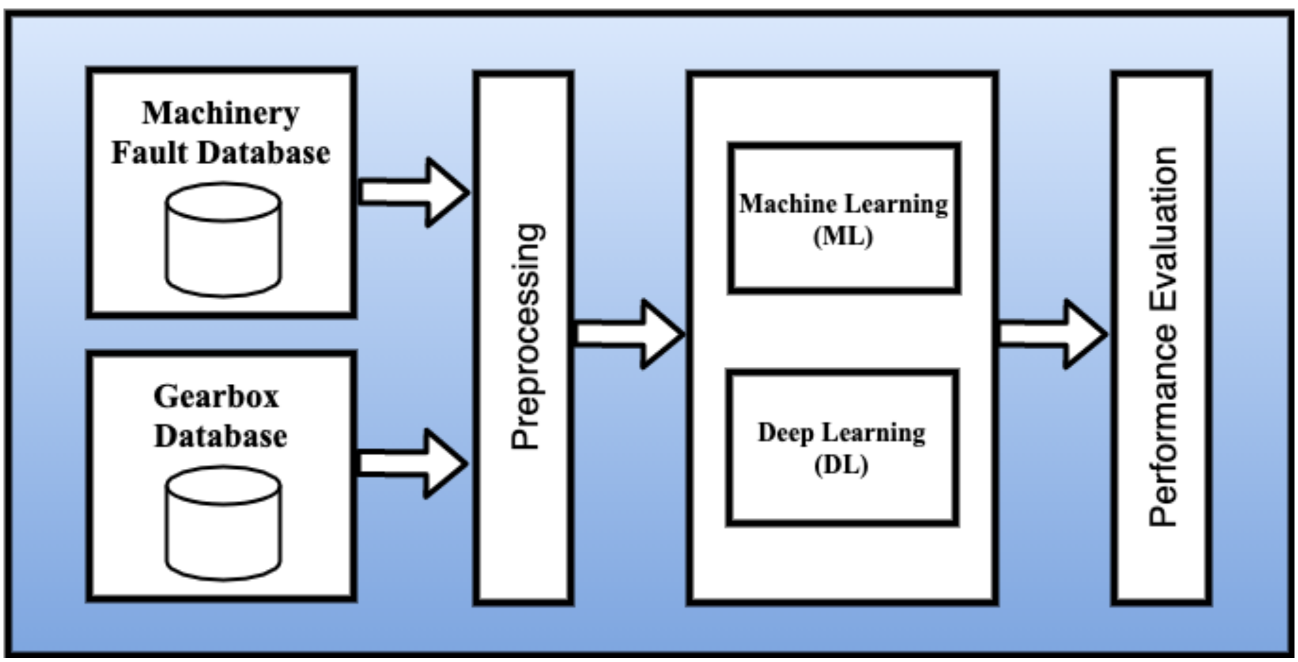

Figure 5.1: Machine learning and deep neural network pipeline for gearbox and rotatory machinery.

5.1 Raw data / Sensors reading

The data from the gearbox have been collected by using four vibrations sensors as shown in **Figure 5.2**. The operating frequency used by sensors is 30Hz. These readings from the sensors are taken by varying load 0 to 90 percent and stored in the database. The gearbox database contains information about the health condition of the gearbox such as

- Broken teeth
- Normal

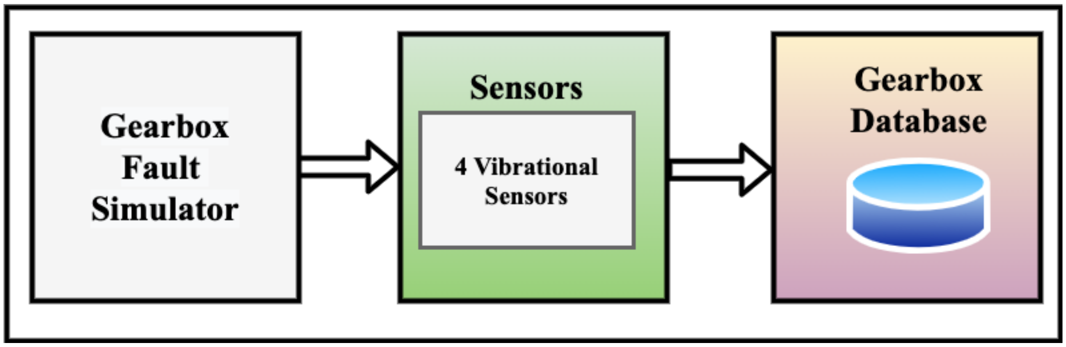

Figure 5.2: Gearbox data acquisition

The data stored in the machinery fault database is acquired with the help of six accelerometers, a microphone, and a tachometer attached to the machine fault simulator [4] as shown in **Figure 5.3**. It contains a total of 1951 scenarios with different operating conditions and loads**.** The data describe the normal and five faulty states of the rotatory machine.

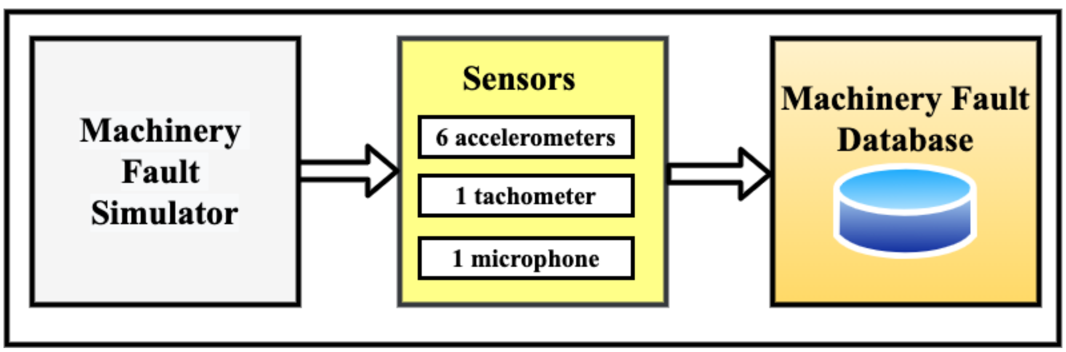

Figure 5.3: Rotatory machinery data acquisition

5.2 Preprocessing

It is an important step in any kind of analysis. During the preprocessing step, raw data is quality checked, trimmed, or cleaned to remove any bias in the data. The data coming from the databases is pre-processed by first doing the quality check where we check the missing (NAN) values (**Figure 5.4**). If the missing values are found, it is imputed with the mean value. In the next step the standard deviation of the dataset is performed and then labelled the data by categories (binary or multi class) specific to that dataset. Finally, the labelled dataset is merged into the single file containing all the required information.The data preprocessing helps us to

- Improve the quality of data

- Checking missing values
- Clean the data
- Normalized the data
- Transforming the data into the required format
- Find the outlier or noisy data before applying any ML or DNN model

5.2.1 Standard Deviation

It shows the spread of the data distribution by calculating the distance between each data point and the mean. It is typically had two forms

- Population standard deviation
- Sample standard deviation

The only difference between these is in the case of population, the standard deviation for the whole population is calculated by dividing the data points by N, and in the case of the sample, standard deviation from the number of samples is calculated by dividing the number of data points in the sample i.e., N-1 [26].

.

$$S_x = \sqrt{\frac{\sum(X_i - X)^2}{N-1}}$$

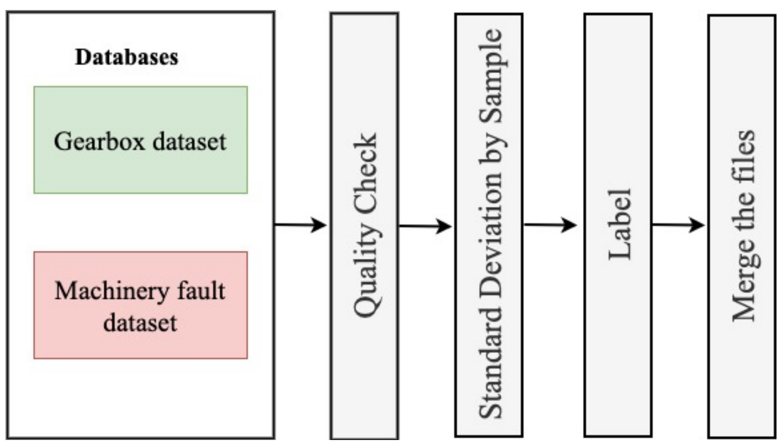

Figure 5.4: Preprocessing pipeline

5.3 Machine Learning Pipeline

In our study, we are dealing with the classification problem and our data are labeled so that is why we used supervised learning techniques. There are many supervised learning algorithms used to solve classification problems, but we used these algorithms

- Decision Tree
- Random Forest
- Adaboost (Adaptive Boosting)

When we applied the ML model to the gearbox and machinery faults study, our initial goal was to learn and test the different types of ML algorithms. Therefore, we selected only those algorithms that minimized the type 1 and type 2 errors as minimum as possible. Another reason for using decision trees and random forests was that they can be used for classification and regression problems.

5.3.1 Decision tree

The decision tree has low bias and high variance. It means the model performs very well on the training dataset and its performance was drop-down on the test dataset. Sometimes it leads to the overfitting problem. Although it is simple and easy to implement.

5.3.2 Random Forest

It can be used for both classification and regression problems. To overcome the problem of high variance in the decision tree, it is good to have used multiple decision trees instead of a single tree. So that is why the random forest is used to overcome the overfitting problem in the decision tree. It is also easy to tune its hyperparameter such as the number of trees in the forest etc. The tree in the forest was created up to its complete depth.

5.3.3 Ada-boost (Adaptive Boosting)

It is an ensemble technique that uses an iterative learning approach to turn the weak learning classifier into the strong classifier by learning the mistakes of the previous model. It used a sequential learning approach instead of a parallel learning approach in a random forest. Stamps (a tree with a single depth) are used to create the decision tree.

5.4 Deep Neural Network (DNN) Pipeline

The goal of using DNN pipeline is to improve the efficiency of the model on given datasets. Relu activation function is used at the input, and hidden layers and sigmoid is used at the output layer as shown in the **Figure 5.5.** Different neurons were used in each layer. This combination of neurons given us the desired results.

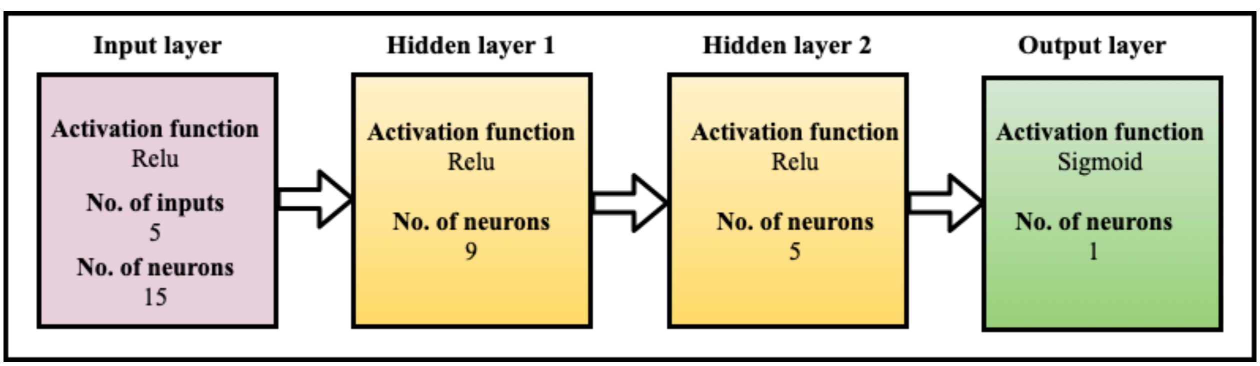

Figure 5.5: DNN architecture for gearbox

The DNN model of rotatory machines contains two hidden layers, one input and output layer. Relu activation function is used in the input, and hidden layers and softmax is used at the output layer as shown in **Figure 5.5.**

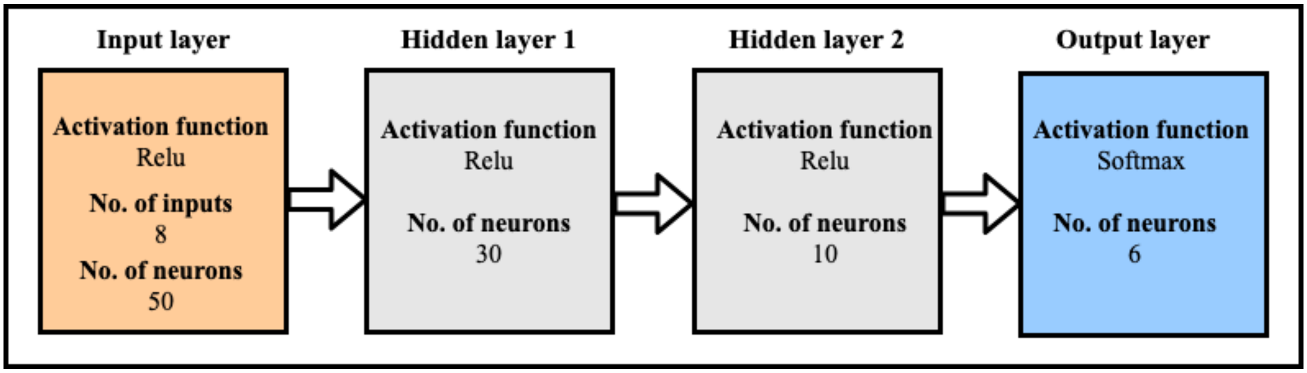

Figure 5.5: DNN architecture for rotatory machine

5.4.1 Activation Function

The activation function is a key element of neural networks; it determines whether to activate or not the neuron. All the hidden layers of neural network behave like a linear function without the activation function. Following activation functions are used in this study

- ReLU
- Sigmoid
- Softmax

The reason for using relu activation function was to avoid the vanishing and exploding gradient problem during backpropagation. Gearbox study is binary classification problem, that is why we used the sigmoid activation function at the output layer, and rotatory machinery fault study, we have a multi-classification problem, softmax was used at the output layer.

1. ReLU

It is the most popular non-linear activation function in MLP and DL. It transforms all the negative values to zero and the positive values remain the same.

$$f(x) = \begin{cases} 0, & if\ x \leq 0 \\ 1, & if\ x > 0 \end{cases}$$

2. Sigmoid

It will transform input between 0 and 1 and is a good choice for binary classification problems.

$$f(x) = \frac{1}{1 + e^{-x}}$$

3. Softmax

It is used for multi-classification problems. Softmax assures that the total probabilities of all our outputs are equal to one.

$$\sigma(xj) = \frac{e^{x_j}}{\sum_i e^{x_i}}$$

5.5 Performance Evaluation

The performance of the models is evaluated by using the following techniques

- Confusion matrix
- Accuracy
- Error Rate (ERR)
- F1 Score
- True Positive Rate (TPR)
- False Positive Rate (FNR)
- Area Under Curve (AUC) Score
- Receiver Operating Characteristic (ROC) curve
- Mean Squared Error

5.5.1 Confusion matrix

It is a simple method to evaluate the performance of the classification models. The matrix describes how many classes were predicted correctly and incorrectly predicted. It is used to evaluate the result of the predicted model with the class outcome to see the number of the classes that were correctly classified [27, 28]. These are the key term used in the confusion matrix **Figure 5.6.**

- **True Positive (TP):** Correctly predicted the class as 'positive' when the actual class is also positive.
- **False Positive (FP):** Incorrectly predicted the class as 'positive' when the actual class is negative. It is also called type I error
- **True Negative (TN):** Correctly predicted the class as 'negative' when the actual class is also negative
- **False Negative (FN):** Incorrectly predicted the class as 'negative' when the actual class is positive. It is also called a type II error.

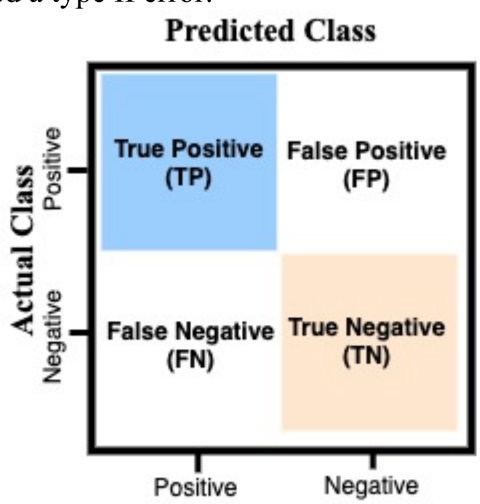

Figure 5.6: Confusion matrix

5.5.2 Accuracy

It is calculated by the number of correct predictions divided by the total number of the dataset. The higher the value of accuracy means better the performance of the model.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

5.5.3 Error Rate (ERR)

It is calculated by the number of incorrect predictions divided by the total number of the dataset. The value of ERR is between 0 and 1. The '0' means the model has no error and '1' means the worse model.

$$ERR = \frac{FP + FN}{TP + TN + FP + FN}$$

We can calculate the error rate also as

$$ERR = 1 - Accuracy$$

5.5.4 True Positive Rate (TPR)

It is used to measure the ability of a model to detect true values. It is also called recall or sensitivity.

$$TPR = \frac{TP}{FN + TP}$$

5.5.5 False Positive Rate (FPR)

Negative cases were incorrectly identified as a positive case in FPR. The good model has an FPR very low.

$$FPR = \frac{FP}{TN + FP}$$

5.5.6 Precision

It is the ratio of predictive positive to true positive.

$$Precision = \frac{TP}{FP + TP}$$

5.5.7 F1-Score

In the case of balance class, accuracy is a suitable choice to evaluate the performance of the model, but for imbalance classes, this approach does not work. The F1 score is a better choice to evaluate the performance of imbalanced datasets. Higher the value of F1 the better the performance of the model. The value of the F1 score is between '0' and '1'.

$$F1 - score = \frac{2 * Precision * Recall}{Precision + Recall}$$

5.5.8 Mean Squared Error (MSE)

MSE error is the difference between the actual output and predicted output divided by the total number of data points as shown in the equation

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_{actual-} - Y_{Predicted})$$

5.5.9 AUC Score

The higher the value of AUC, the better the model has separability. The minimum value of the AUC score is 0 and the maximum value is 1. The ideal condition is when TP and TN are separate from each other, and the AUC score is 1 as shown in **Figure 5.7**.

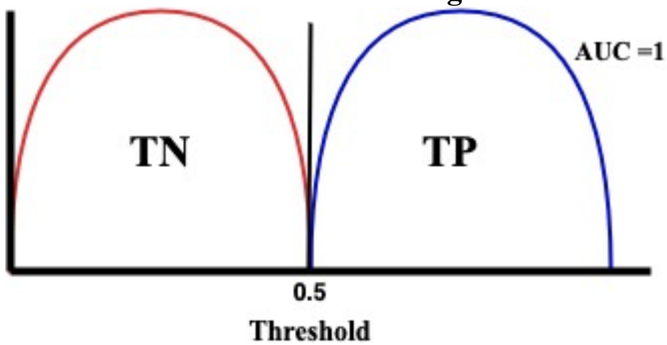

Figure 5.7: AUC Score

5.5.10 ROC Curve

The ROC curve was plotted with FPR and TPR. The smothered the curve the better the model is

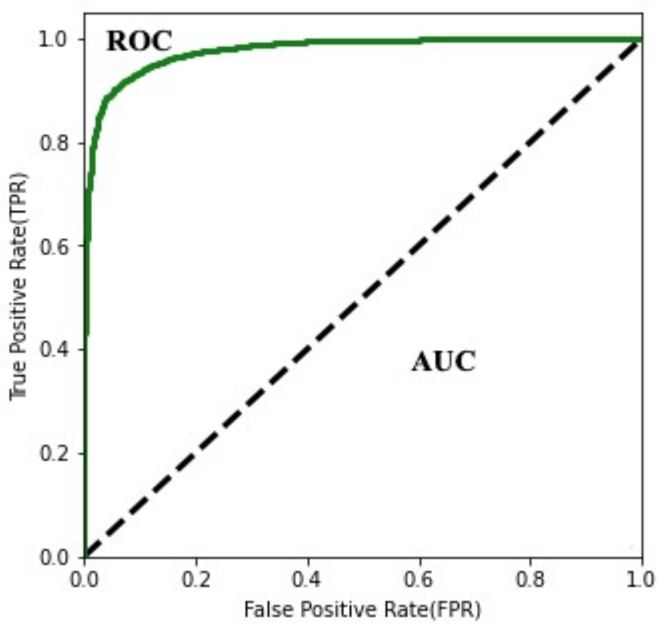

Figure 5.8: ROC curve

6 Results

Training and test data are required to build and validate the results of the machine learning (ML) and deep neural network (DNN) model. Here we have analyzed two datasets i.e., gearbox and machinery fault studies, which are further divided into training and test datasets to build and evaluate the performance of ML and DNN models. The models are learned from the training set and performance is evaluated on test data or unseen data. In both studies, seventy percent of the data is used for training and thirty percent is used for testing the models as shown in **Figure 6.1**.

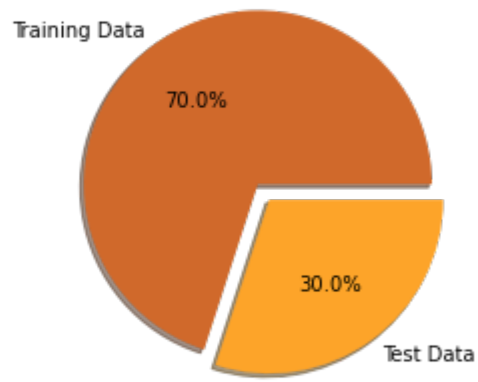

Figure 6.1: Ratio of training and test data to train and evaluate the performance of the ML and DNN model.

6.1 Gearbox Fault Prediction

The gearbox fault prediction dataset (n=4000000) consists of only two classes: normal and broken teeth. It is a binary classification problem. The training data contains 2800000 records (70 %) and the test data contains 120000 records (30 percent). The records are equally distributed among the classes. This means we have a balanced classification problem **Figure 6.2**.

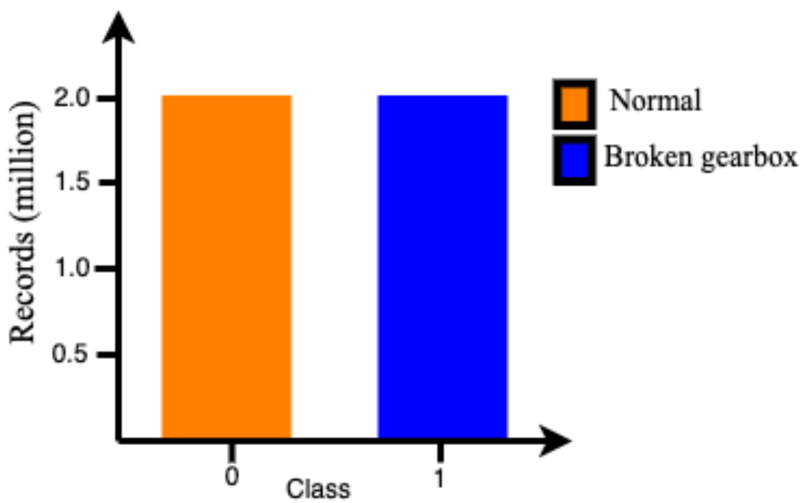

Figure 6.2: Data distribution among normal and broken gearbox classes.

The information from the descriptive features was obtained with the help of sensors. All the readings from the sensors were numerical. This is a binary classification problem, where 0 means normal class and 1 means broken gearbox teeth class. We have 5 descriptive and one target feature shown in **table 6.1**.

Table 6.1. Descriptive and target features.

Descriptive Features					**Target Features Class**
Vibration Sensor 1 reading (S1)	Vibration Sensor 2 reading (S2)	Vibration Sensor 3 reading (S3)	Vibration Sensor 4 reading (S4)	Load Variation (0 – 90) percent	Binary Classification • Normal: 0 • Broken: 1

6.1.1 Performance Evaluation on raw data

Table 6.2 and **Figure 6.3** describes the results of ML and DLL models on the raw data of the gearbox dataset. We first evaluated and compared the performance of different machine learning models ML (**Figure 6.3 (a-c)**) and DL model (**Figure 6.3d**) using the gearbox raw data i.e without applying any normalization techniques. It means that models were first directly deployed on raw data.

Our results showed that type 1 error was highest in the random forest (RF) (**Figure 6.3c**) and type II was lowest. The DNN model showed that the type 1 error was lowest while the type II error was highest (**Figure 6.3d**).

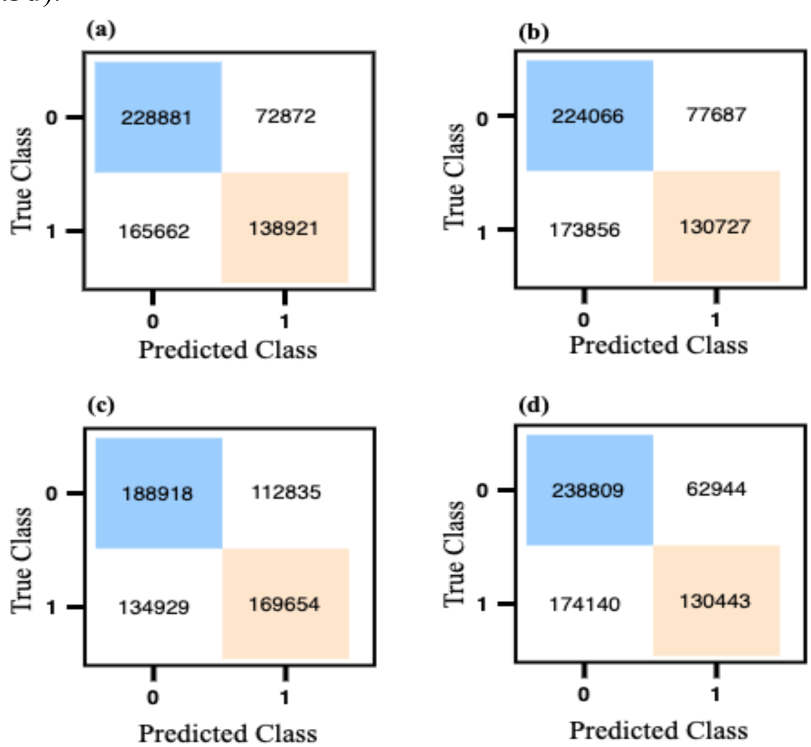

Figure 6.3. Performance evaluation of ML and DLL models on raw data. In each confusion matrix predicted class (x-axis) and true class (y-axis) representing the normal class with '0' and broken gearbox tooth class '1'. Confusion matrix of (a) decision tree (b) AdaBoost (c) random forest (d) deep neural network.

Overall performance of DNN and decision tree is better based on highest accuracy rate/precision rate and lowest error rate than the random forest and AdaBoost (**Table 6.2).** However, the F1-score was best for the RF model.

Table 6.2: Summary statistics of performance of ML and DLL models on raw data.

Model	Accuracy (%)	Precision	Recall	F1-score	Error Rate (%)	MSE
Decision Tree	60.06	0.6559	0.456	0.5380	39.94	0.229
Random Forest	59.13	0.6005	0.557	0.5779	40.87	0.239
Ada boost	58.51	0.6272	0.421	0.5096	41.49	0.415
DNN	60.57	0.6574	0.444	0.5336	39.43	0.386

We next compared the graphical description of the ML and DNN classifier performance on raw data using the ROC. Our results show that the area under the ROC curve of the DNN model is higher than the ML model (RF, DT, and AdaBoost) models (**Figure 6.4**). Based on the ROC curve and AUC score performance of DNN and Random Forest is much better than the decision tree and AdaBoost (**Figure 6.4 a-d**).

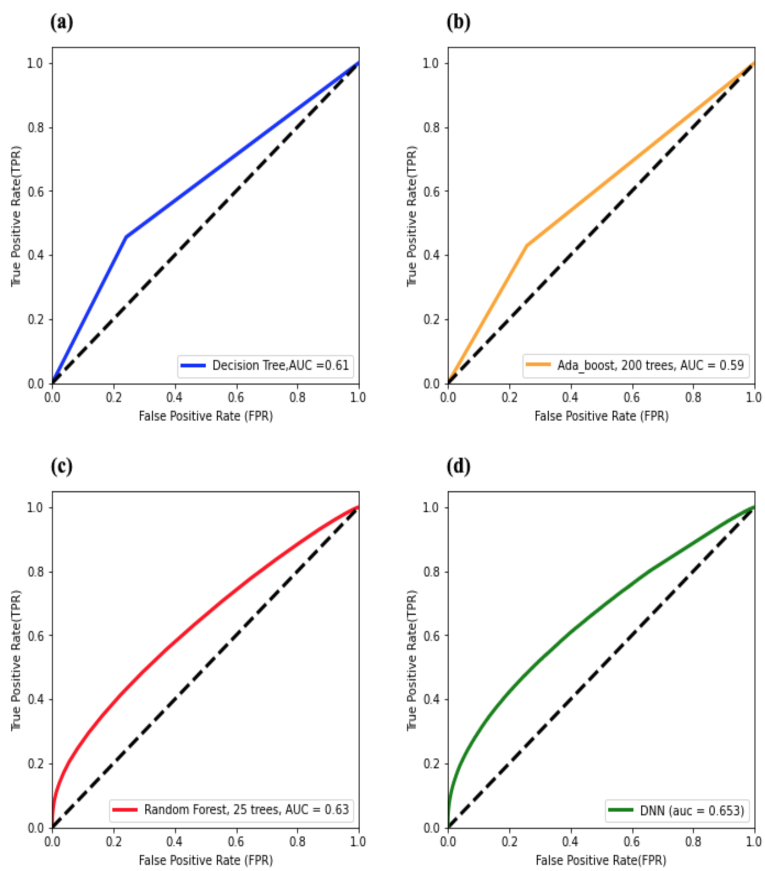

Figure 6.4. ROC curve and corresponding AUC score of both ML and DNN model on raw data (a) decision tree (b) AdaBoost (c) random forest (d) DNN.

6.1.2 Performance Evaluation of normalized data

The standard deviation with sample size (N=10,25,50,100 and 500) used to normalized the gearbox and machinery fault datasets. The motivation for using these sample values was to reduce the error rate and increase the performance of ML and DNN model.

- **Normalized data with sample size N=10**

  We next evaluated the performance of different models using the normalized data by taking the sampling size of N=10. Here instead of directly taking the raw data from the sensor reading, we first take standard deviations for each of the sample sizes 'N=10', and then the models were deployed on this normalized dataset.

  Based on this approach, our results showed that the overall performance of all the models, both the ML and DNN was improved by approximately 10% as compared to the raw data (**Figure 6.5 (a-d), Table 6.3**). The accuracy rate of the DNN model was improved from 60% in raw data to 73% in normalized data (**Table 6.3**).

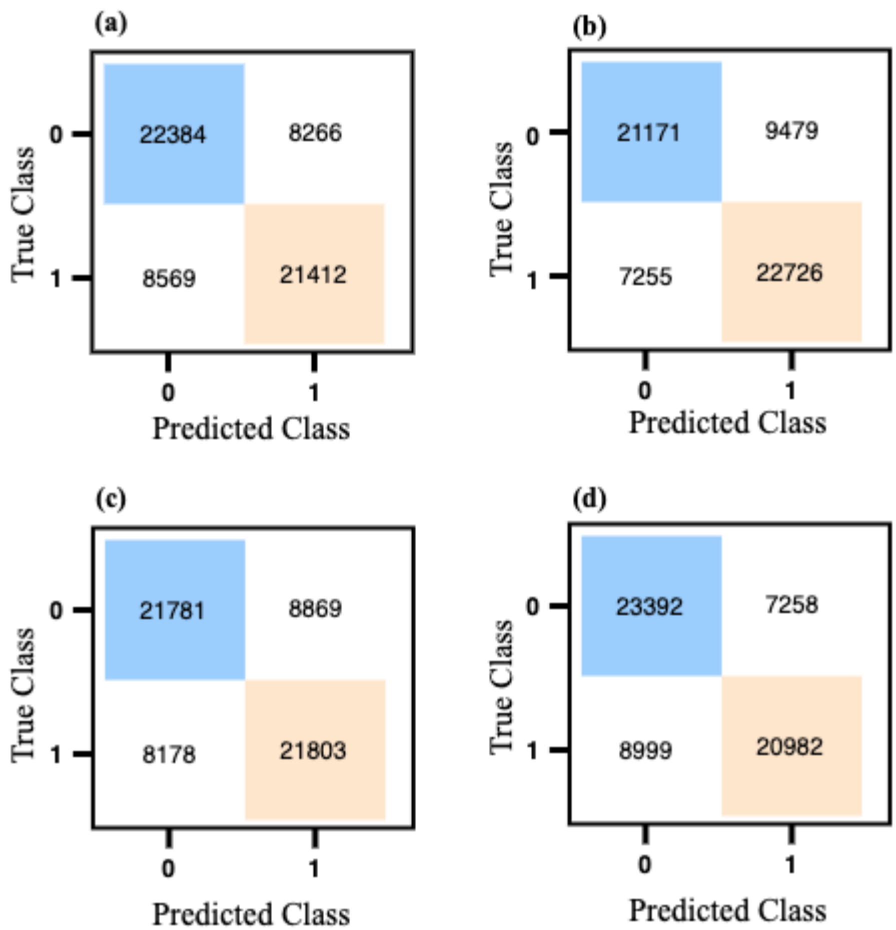

Figure 6.5. Performance evaluation of ML and DLL models with N=10. In each confusion matrix predicted class (x-axis) and true class (y-axis) representing the normal class with '0' and

broken gearbox tooth class '1'. Confusion matrix of (a) decision tree (b) AdaBoost (c) random forest (d) DNN.

Table 6.3. Summary statistics of performance of ML and DLL models on normalized data with N=10.

Model	Accuracy (%)	Precision	Recall	F1-score	Error Rate (%)	MSE
Decision Tree	72.23	0.7215	0.7142	0.7178	27.76	0.182
Random Forest	71.88	0.7108	0.7272	0.7189	28.12	0.182
Ada boost	72.40	0.7056	0.7580	0.7309	27.60	0.276
DNN	73.44	0.7215	0.7553	0.7380	26.56	0.265

The ROC and AUC were significantly improved for all the models using the normalized data with the sampling size N=10 as compared to raw data (**Figure 6.6(a-d)**). Interestingly, we observed that the AUC for the DNN model improved to 0.82 in normalized data compared to raw which was 0.65.

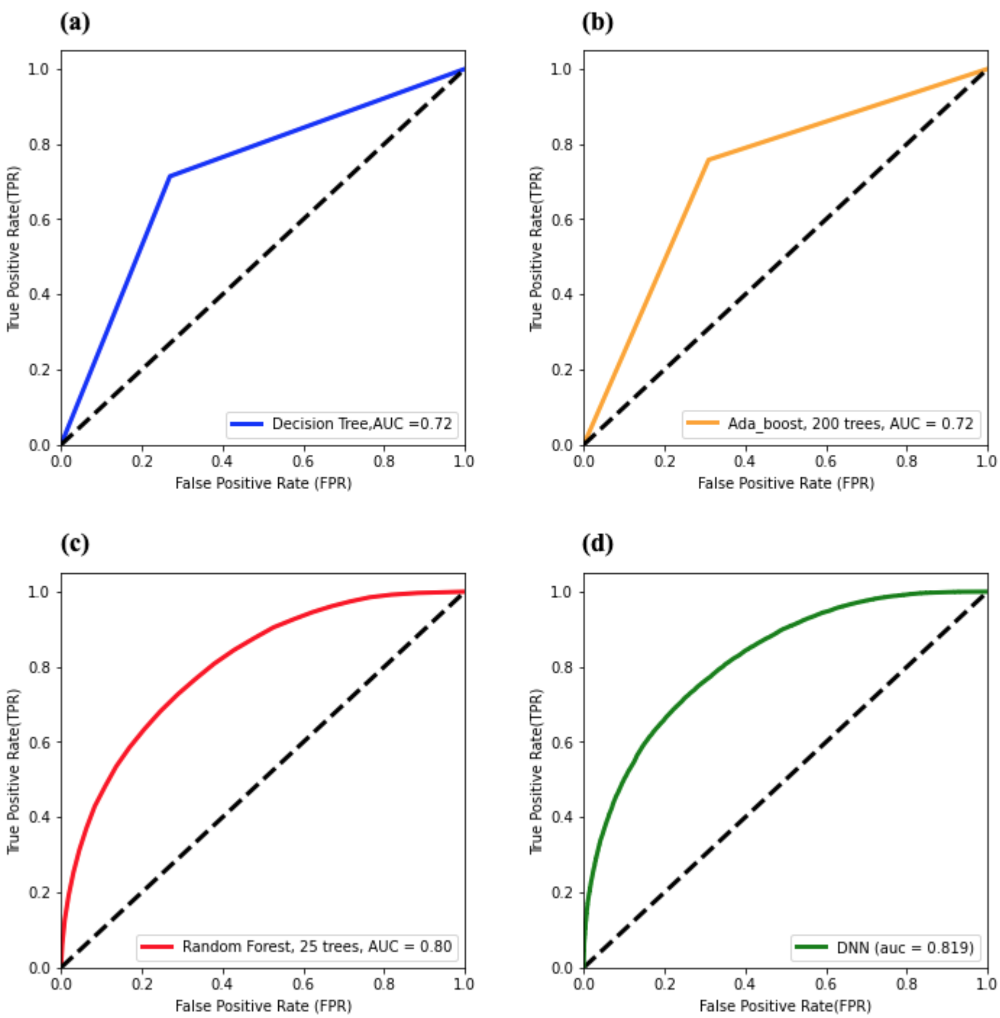

Figure 6.6. ROC curve and corresponding AUC score of both ML and DNN model with N=10 (a) decision tree (b) AdaBoost (c) random forest (d) DNN

- **Normalized data with sample size N=25**

  The performance of the ML and DNN models was evaluated by taking the normalized data with the sampling size of N=25. Type 1 and type II were significantly reduced (**Figure 6.7(a-d)**) as compared to raw data and normalized N=10. The performance of the DNN and RF model is much better as compared to the raw data and N=10 with the accuracy rate approaching 81% in the DNN model.

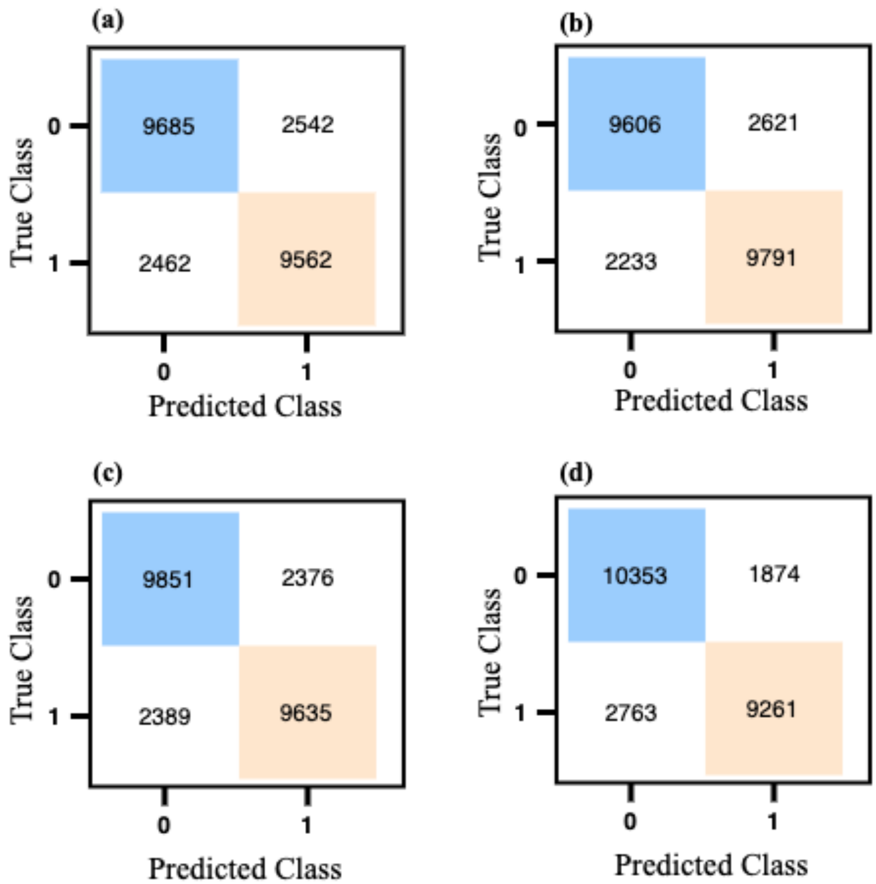

Figure 6.7. Performance evaluation of ML and DLL models with N=25. In each confusion matrix predicted class (x-axis) and true class (y-axis) representing the normal class with '0' and broken gearbox tooth class '1'. Confusion matrix of (a) decision tree (b) AdaBoost (c) random forest (d) DNN.

Table 6.4. Summary statistics of performance of ML and DLL models on normalized data with N=25.

Model	Accuracy (%)	Precision	Recall	F1-score	Error Rate (%)	MSE
Decision Tree	79.36	0.7897	0.7955	0.7926	20.64	0.144
Random Forest	80.35	0.8021	0.8013	0.8017	19.65	0.134
Ada boost	79.98	0.7888	0.8142	0.8013	20.02	0.200
DNN	80.55	0.7936	0.8212	0.8072	19.45	0.192

The AUC of both the RF and DNN model further improved to 0.89 compared to the previous sample size (N=10). It means RF and DNN models' performance was very similar using this sample size N=25 (**Figure 6.8**).

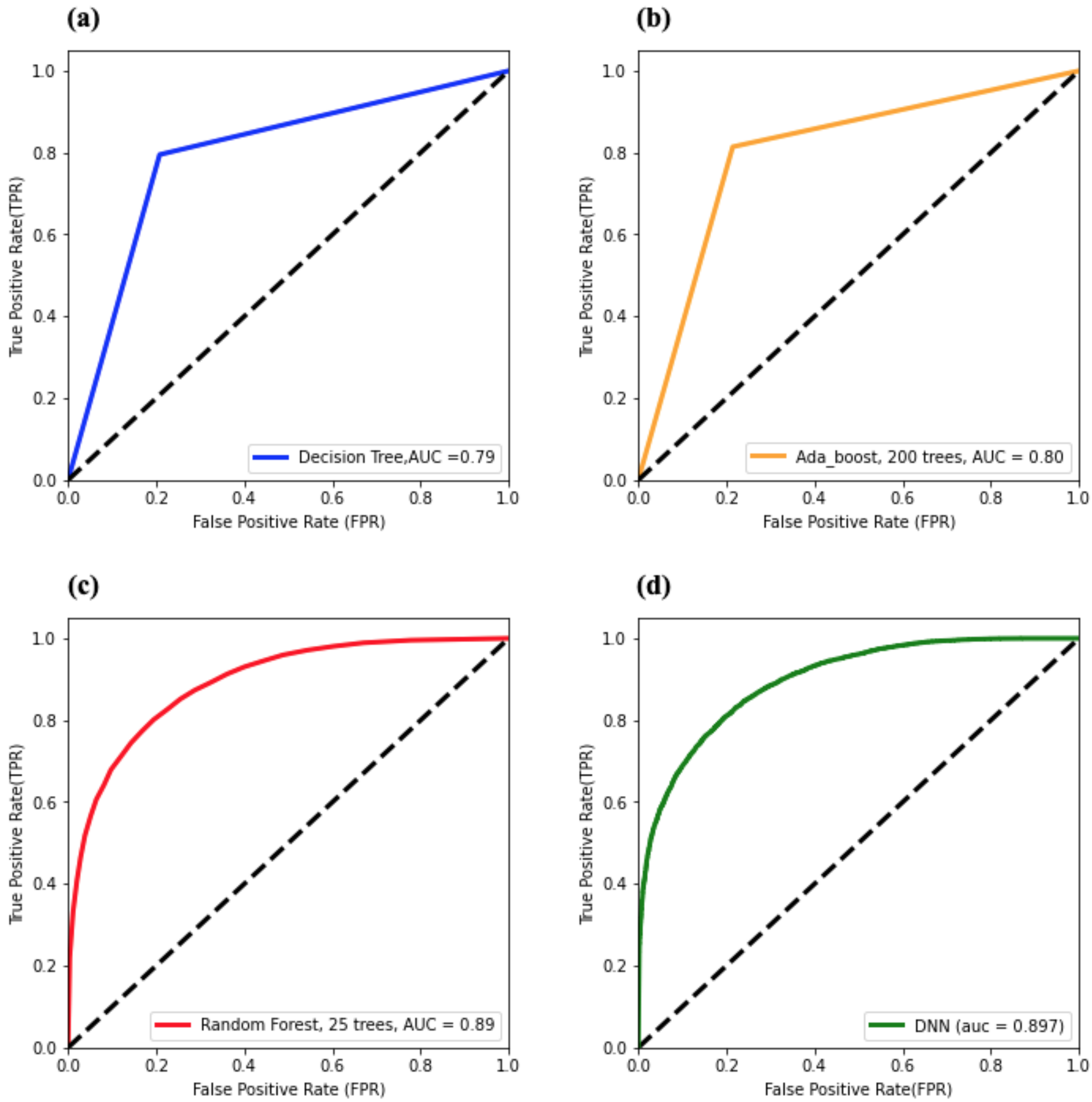

Figure 6.8. ROC curve and corresponding AUC score of both ML and DNN model with N=25 (a) decision tree (b) AdaBoost (c) random forest (d) DNN

- **Normalized data with sample size N=50**

  The performance of the ML and DNN models was further evaluated by taking the normalized data with the sampling size of N=50. Figure 8 and Table 5 describe the results of ML and DLL models on the gearbox dataset. Overall performance of DNN and Random Forest on this is much better than decision tree and AdaBoost looking into the confusion matrix (**Figure 6.9**). The accuracy rate of the DNN and RF model was approx. 86% with the highest precision and F1-scores and the lowest error rate of about 13 % (**Table 6.5**).

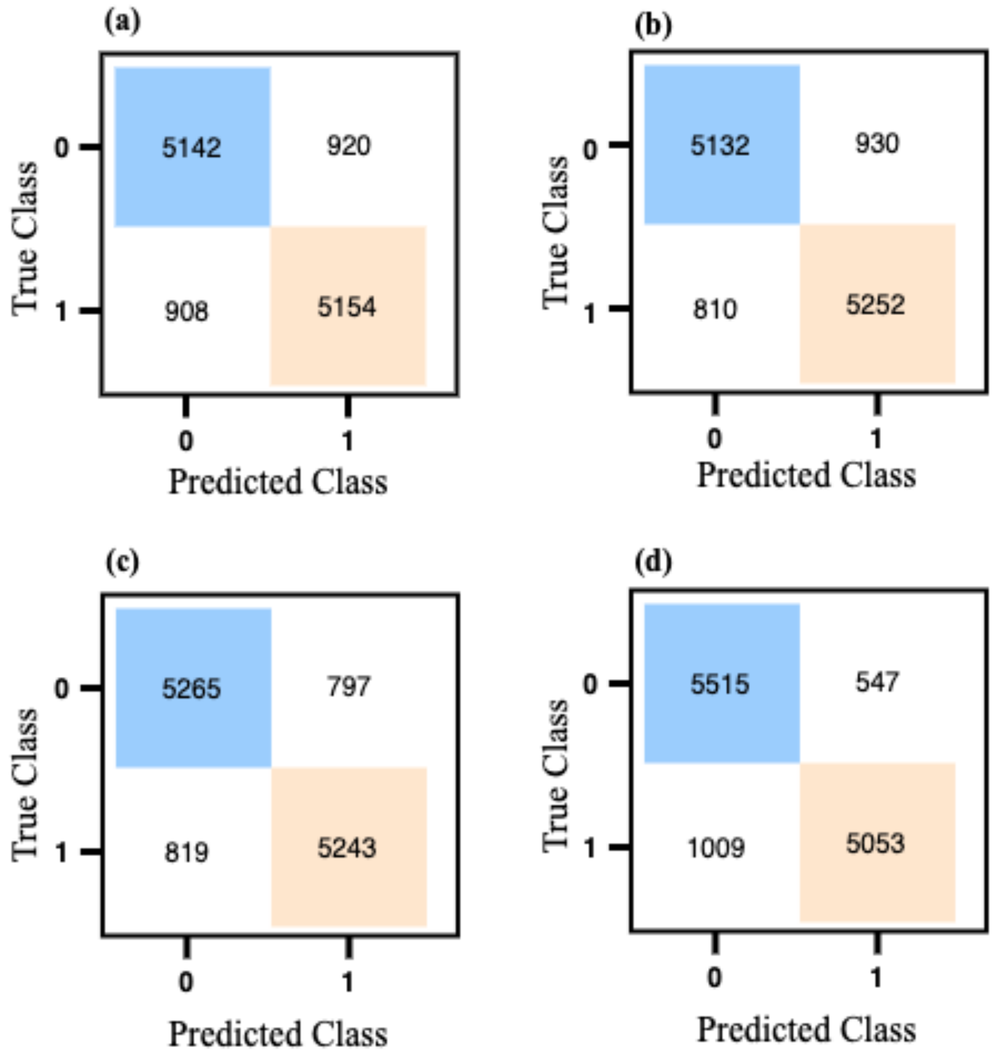

Figure 6.9. Performance evaluation of ML and DLL models with N=50. In each confusion matrix predicted class (x-axis) and true class (y-axis) representing the normal class with '0' and broken gearbox tooth class '1'. Confusion matrix of (a) decision tree (b) AdaBoost (c) random forest (d) DNN

Table 6.5. Summary statistics of performance of ML and DLL models on normalized data with N=50.

Model	Accuracy (%)	Precision	Recall	F1-score	Error Rate (%)	MSE
Decision Tree	84.93	0.8482	0.8508	0.8495	15.07	0.112
Random Forest	86.67	0.8680	0.8648	0.8664	13.33	0.093
Ada boost	85.64	0.8495	0.8663	0.8578	14.35	0.144
DNN	86.54	0.8908	0.8485	0.8692	13.46	0.131

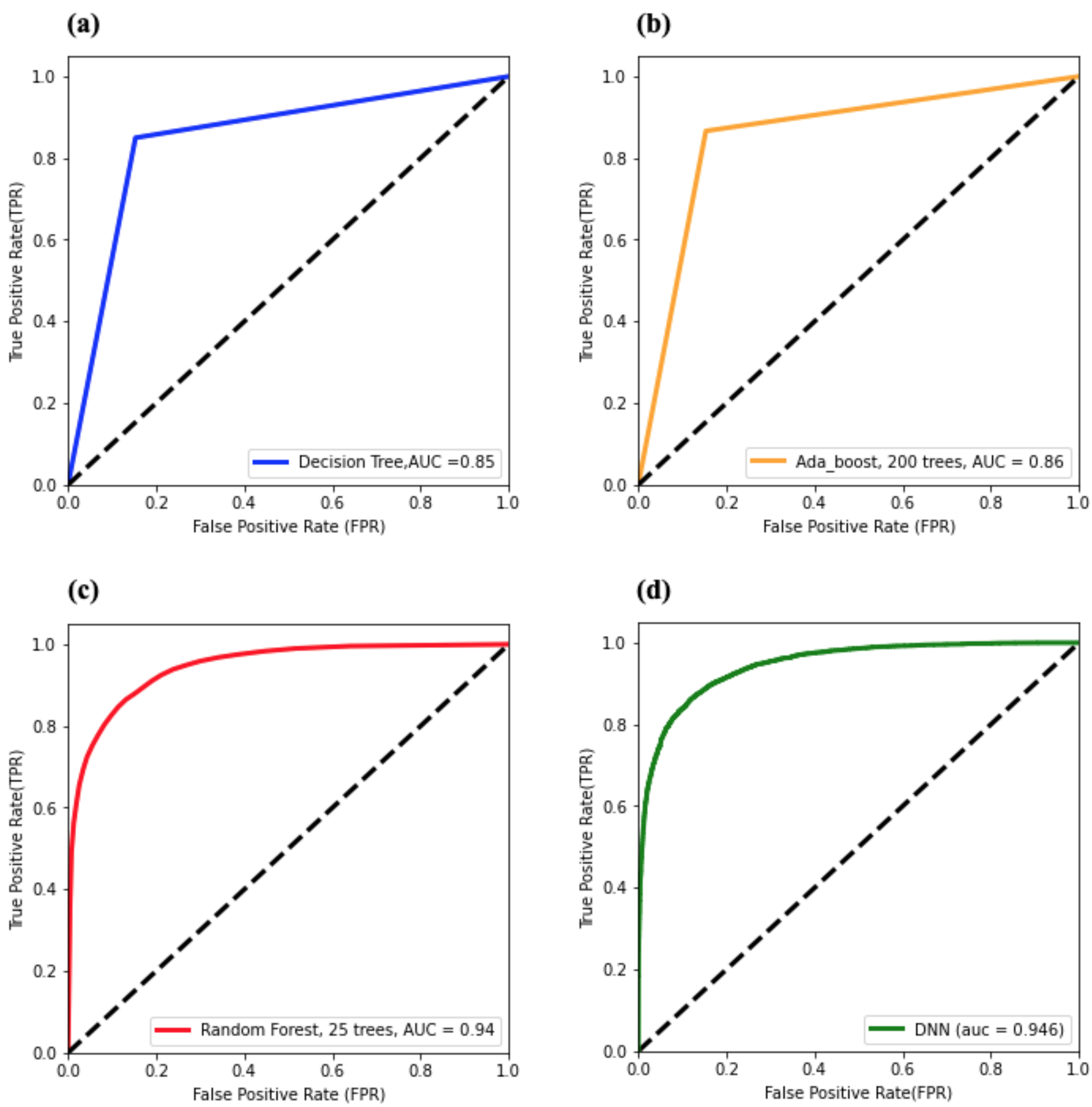

Figure 6.10. ROC curve and corresponding AUC score of both ML and DNN model with N=50 (a) decision tree (b) AdaBoost (c) random forest (d) DNN

The AUC of both the RF and DNN models further improved to 0.94 compared to previously which was 0.89. As previously, the RF and DNN models' performance was very similar using sample size N=50 (**Figure 6.10**).

- **Normalized data with sample size N=100**

  The performance of the ML and DNN models was further evaluated by taking the normalized data with the sampling size of N=100. The overall performance of all models improved remarkably, with the DNN and RF models showing the best results as can be seen in the confusion matrixes (**Figure 6.11**). The accuracy rate of the DNN and RF model reached approx. 93% with the highest precision and F1-scores and the lowest error rate of about 7% (**Table 6.6**).

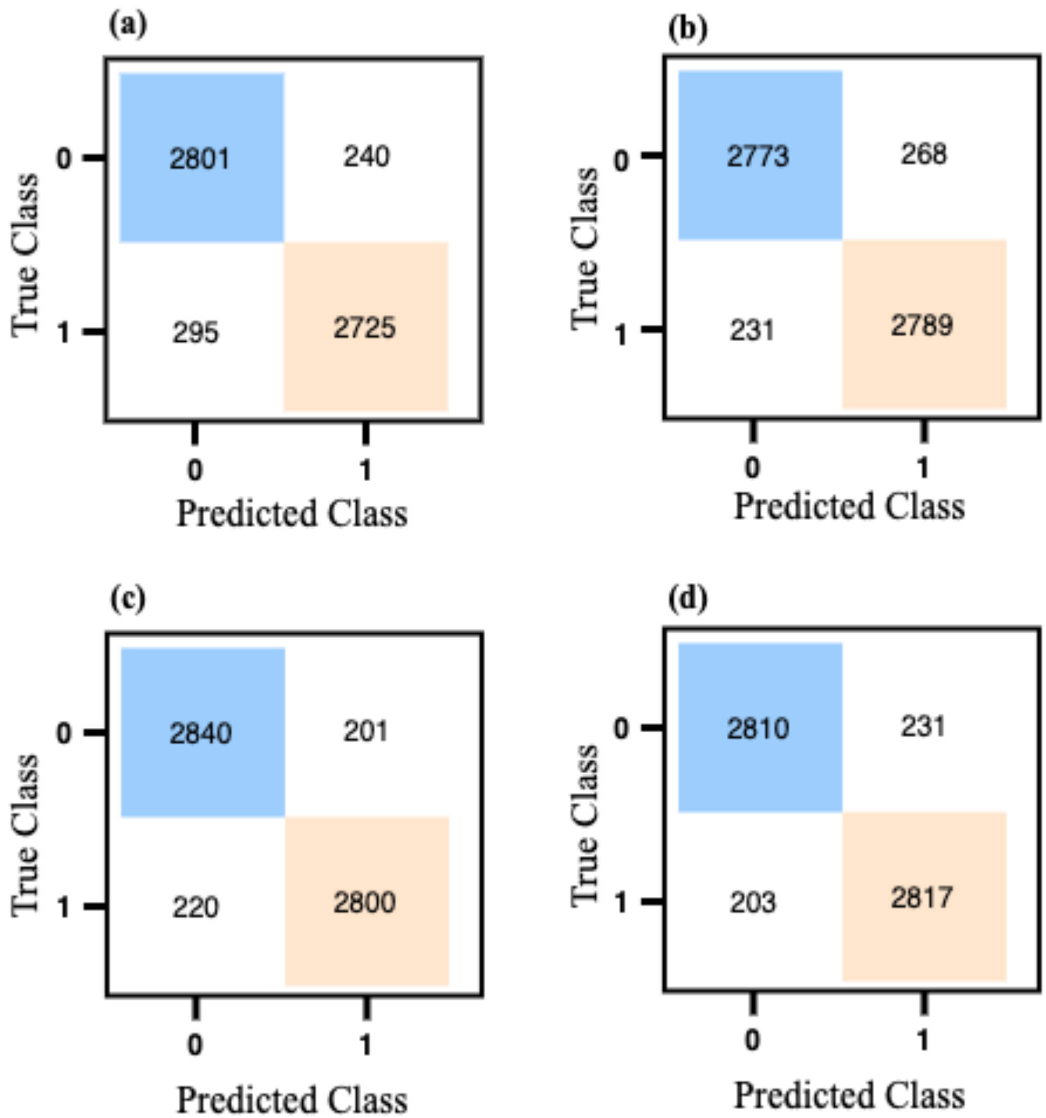

Figure 6.11. Performance evaluation of ML and DLL models with N=100. In each confusion matrix predicted class (x-axis) and true class (y-axis) representing the normal class with '0' and broken gearbox tooth class '1'. Confusion matrix of (a) decision tree (b) AdaBoost (c) random forest (d) DNN

Table 6.6. Summary statistics of performance of ML and DLL models on normalized data with N=100.

Model	Accuracy (%)	Precision	Recall	F1-score	Error Rate (%)	MSE
Decision Tree	91.11	0.9181	0.9023	0.910	8.89	0.074
Random Forest	93.05	0.9331	0.9271	0.930	6.95	0.051
Ada boost	91.76	0.9123	0.9235	0.917	8.24	0.082
DNN	93.21	0.9424	0.9168	0.925	6.79	0.072

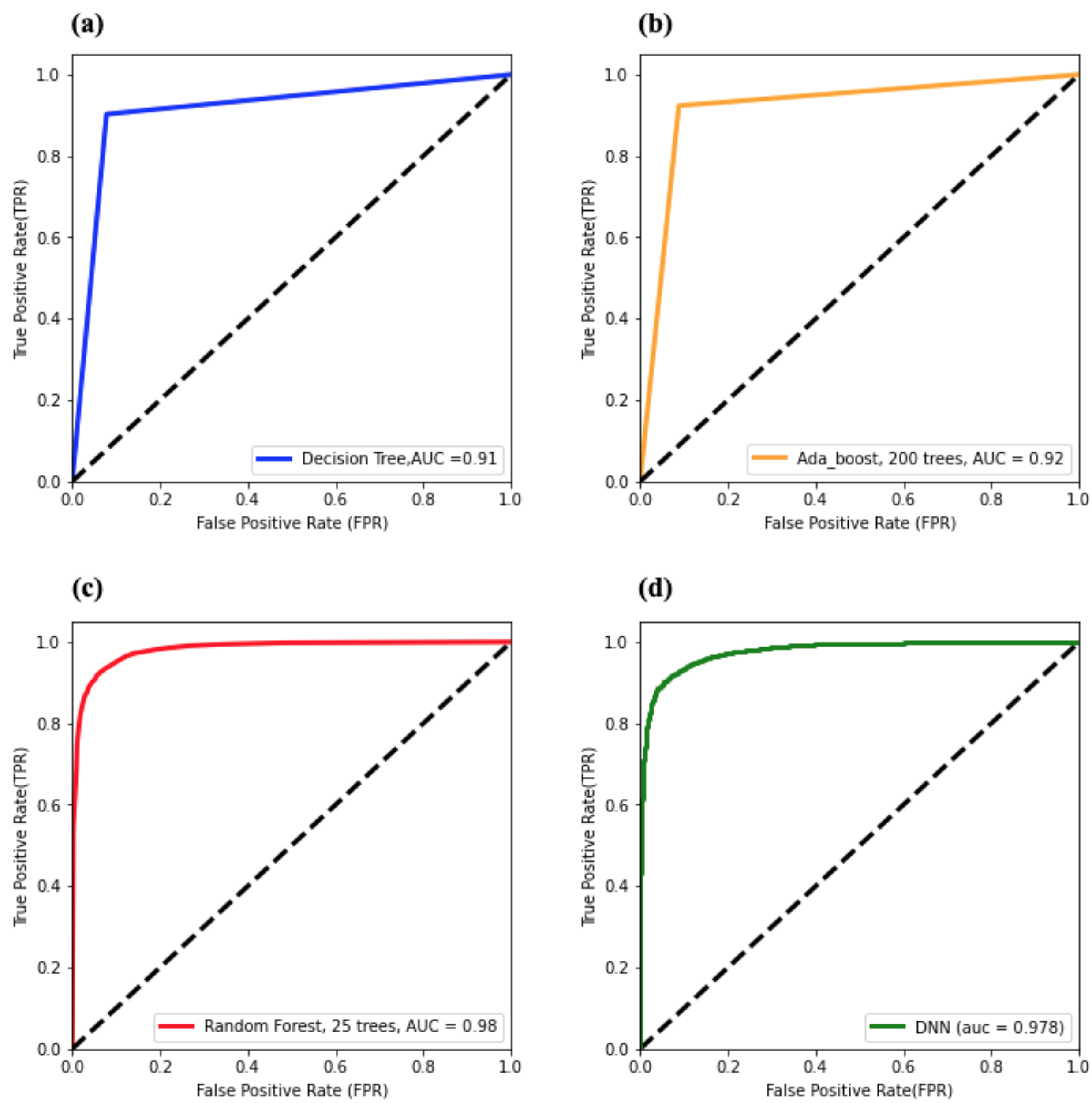

Figure 6.12. ROC curve and corresponding AUC score of both ML and DNN model with N=100 (a) decision tree (b) AdaBoost (c) random forest (d) DNN

The ROC further improved for all models in this case and the AUC of both RF and DNN models was about 0.98. As previously, the RF and DNN models' performance was very similar using sample size N=100 (**Figure 6.12**).

6.2 Machinery Fault Prediction

The second dataset that we analyzed here was the machinery fault prediction (MFP) dataset, which is a multi-classification problem. We have six classes such as normal, imbalance, horizontal misalignment, vertical misalignment, underhang, and overhang bearing faults in this study. The training data contains 6828550 records (70 %) and test data holds 292650 records (30 percent). The distribution of records among the multi-classes in the MFP dataset is shown in **Figure 6.13**.

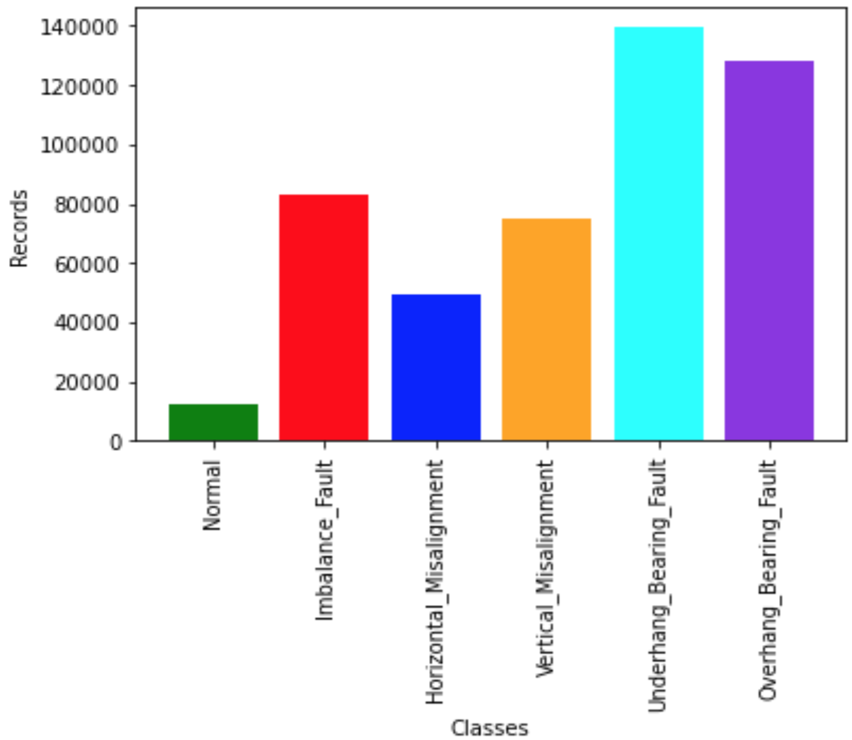

Figure 6.13. The number of records in each of the six classes in MFP.

The records from the descriptive features were obtained with the help of sensors. All the readings from the sensors are numerical. This is a multi-classification problem, where '0' represents normal class, '1' represents imbalance fault class, '2' represents horizontal misalignment, '3' represents vertical misalignment, '4' represents underhang bearing fault and, '5' represents overhang bearing fault classes. We have 8 descriptive and one target feature (**Table 6.7)**.

§

Table 6.7: Descriptive and target features of rotatory machinery dataset. S1 represents reading from the tachometer, S2-S4 represents reading from the underhang bearing accelerometers, S5-S7 represent reading from overhang bearing accelerometers and S8 reading from the microphone.

Descriptive Features								Target Features
S1	S2	S3	S4	S5	S6	S7	S8	Multi classification • Normal: 0 • Imbalance: 1 • Horizontal misalignment: 2 • Vertical misalignment: 3 • Underhang bearing fault: 4 • Overhang bearing fault: 5

The results of both ML and DNN models were evaluated and compared with each other. The performance of the models is evaluated by confusion matrix, accuracy, F1-score, AUC score, and ROC curve. The error was calculated by using MSE.

6.2.1 Performance evaluation of ML model in MFP dataset

This section describes the performance of the ML model on the given datasets. The algorithm we used in ML is random forest. The confusion matrix is shown in (**Figure 6.14**). summarizing the performance of the model. The correctly classified classes using this model were shown diagonally in the confusion matrix (**Figure 6.14**). While other elements (non-diagonal) of the confusion matrix indicate incorrectly classified records. Our results showed that 153 cases of class normal were incorrectly classified into other classes such as 11 records were classified as class imbalance, 90 records in horizontal misalignment, 22 records in vertical misalignment, 13 records in underhang bearing fault, and 17 records in overhang bearing fault (**Figure 6.14**). The confusion matrix helps to analyze different types of errors in classification.

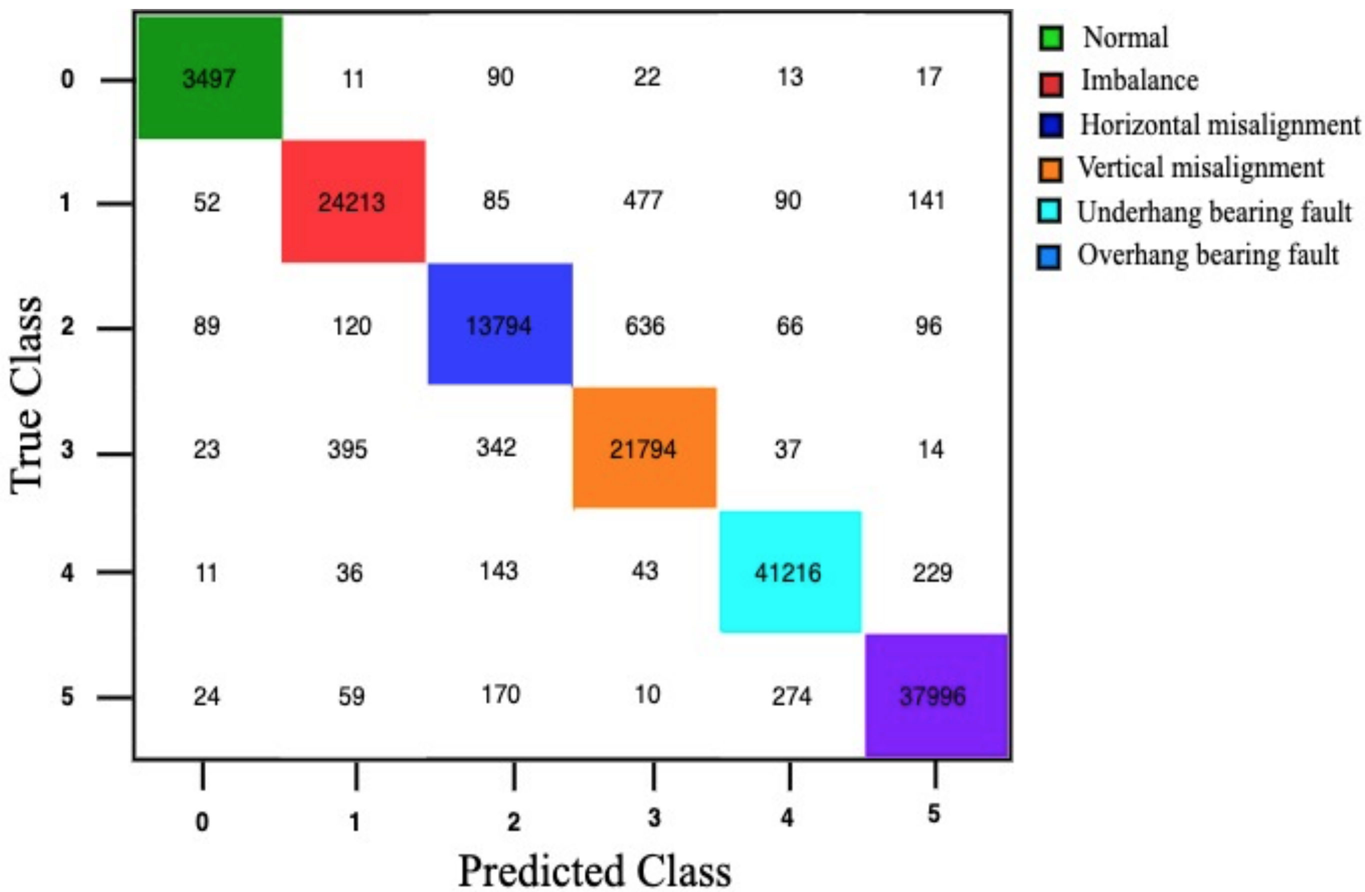

Figure 6.14. Performance evaluation of RF of MFP using a confusion matrix. In confusion matrix predicted class (x-axis) and true class (y-axis) representing classes: normal with '0', Imbalance with'1', horizontal misalignment with '2', vertical misalignment with '3', underhang bearing fault with '4' and overhang bearing fault '5'.

Figure 6.15 illustrates the predictive performance of RF models on multi-classification problems by using the ROC curve while plotting a false positive rate against the true positive rate. The area under the ROC curve of the normal class was 0.98 percent, which means the ML model distinguishes between the class normal with other classes with an accuracy of 98 percent.

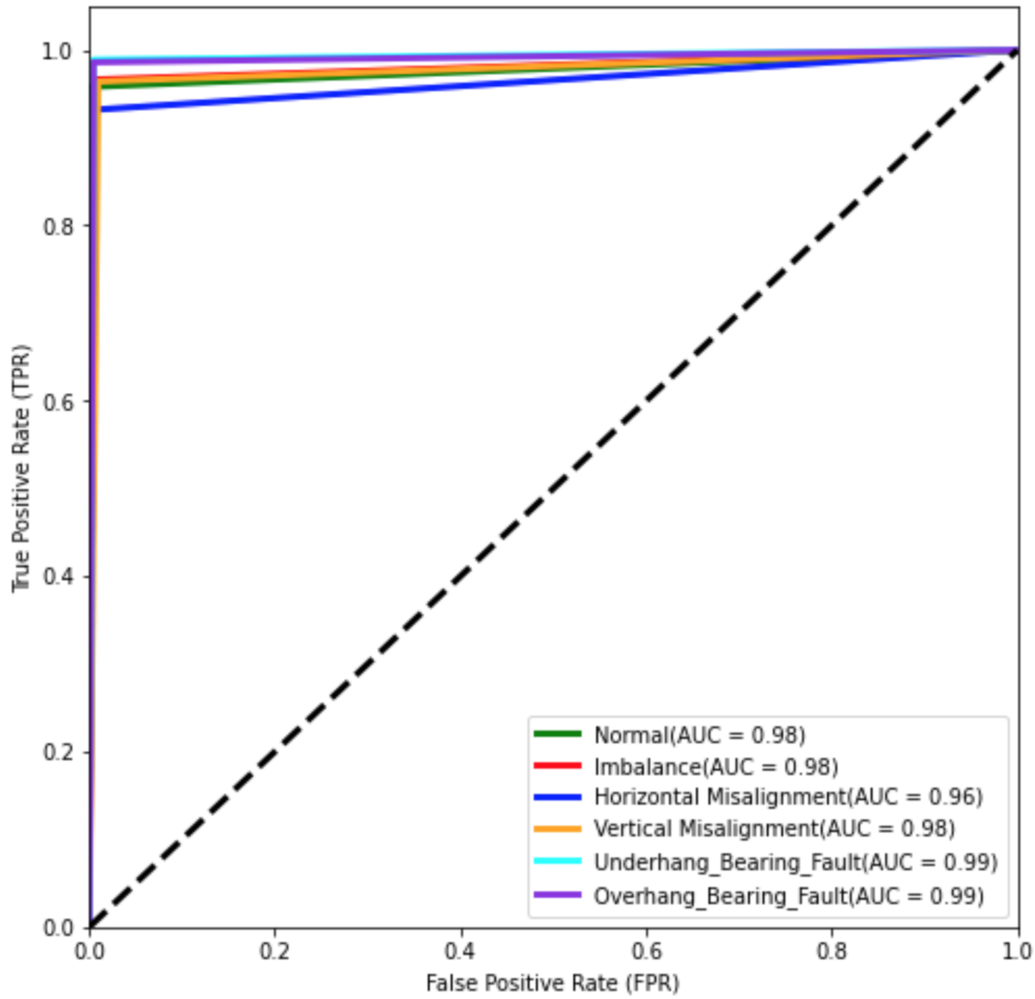

Figure 6.15: ROC curve and corresponding AUC score of RF model in MFP dataset.

6.2.2 Performance evaluation of DNN model in MFP dataset

This section describes the performance of the DNN model using the MFP dataset. The model correctly classified the multi-class using this approach as shown with diagonal numbers in **Figure 6.16**. Our results show that the number of correctly classified records by the DNN model is slightly decreased as compared to the RF model as shown in **Figure 6.16**.

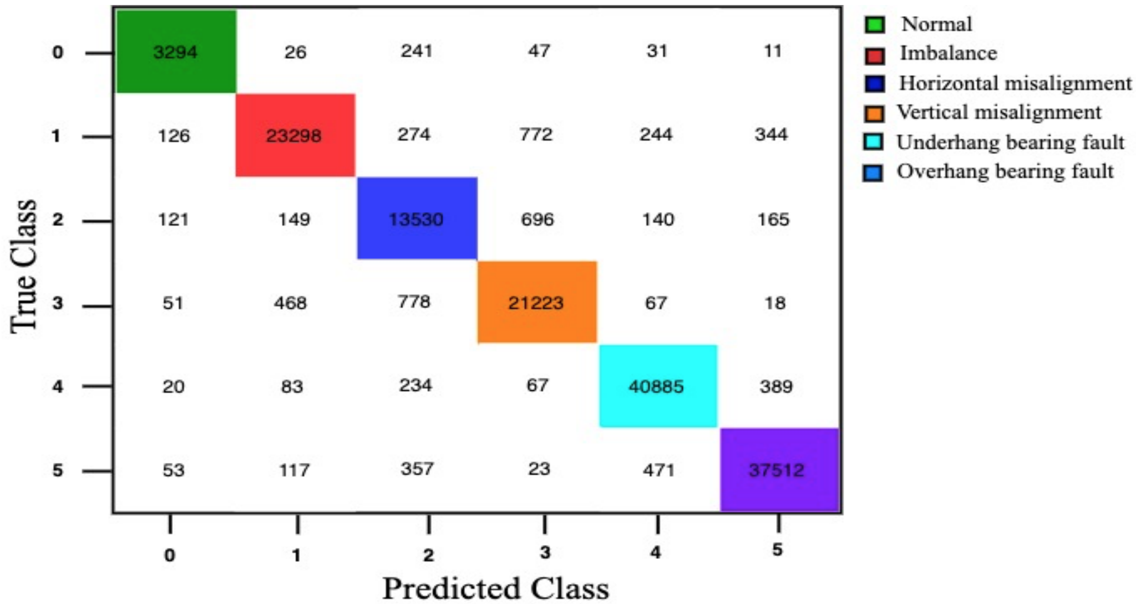

Figure 6.16. Performance evaluation of DNN model in MFP dataset. In confusion matrix predicted class (x-axis) and true class (y-axis) representing multi classes: normal with '0', Imbalance with'1', horizontal misalignment with '2', vertical misalignment with '3', underhang bearing fault with '4' and overhang bearing fault '5'.

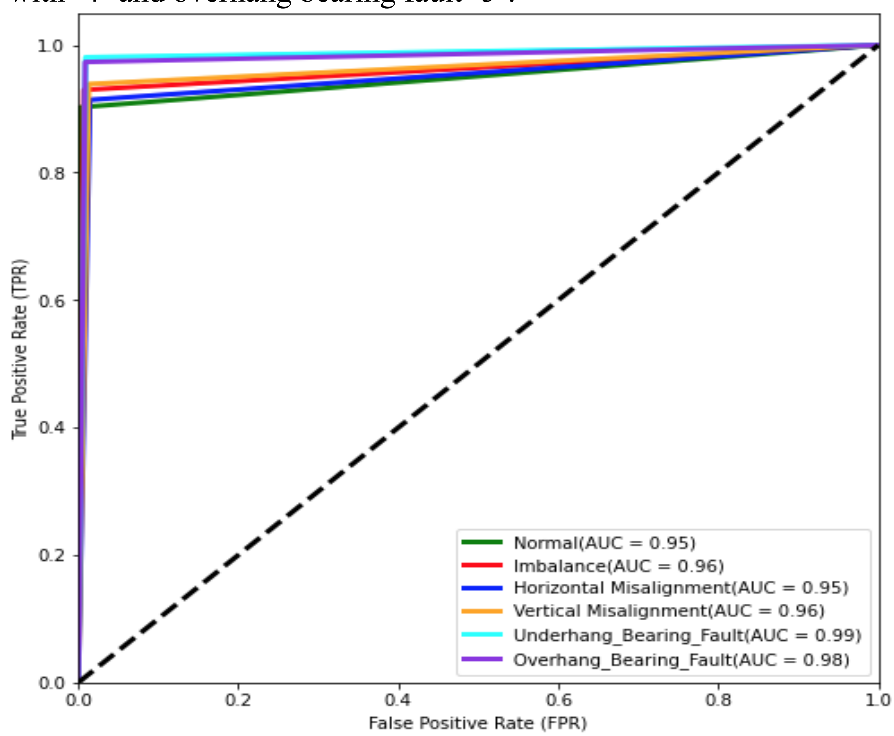

Figure 6.17: ROC curve and corresponding AUC score of DNN model in MFP dataset. The underhang bearing fault AUC score of 0.99 remained the same as in the RF model. While all other classes, AUC has slightly decreased compared to the RF model, showing that the RF model performed better than the DNN model for the MFP dataset (**Figure 6.17**).

In the DNN model, we have also compared the predicted output to the actual output. Here the error was calculated between actual and predicted output. During backpropagation of DNN model weights are updated with each iteration or epochs. The error is reduced, and accuracy is increased with each iteration as shown in **Figure 6.18**.

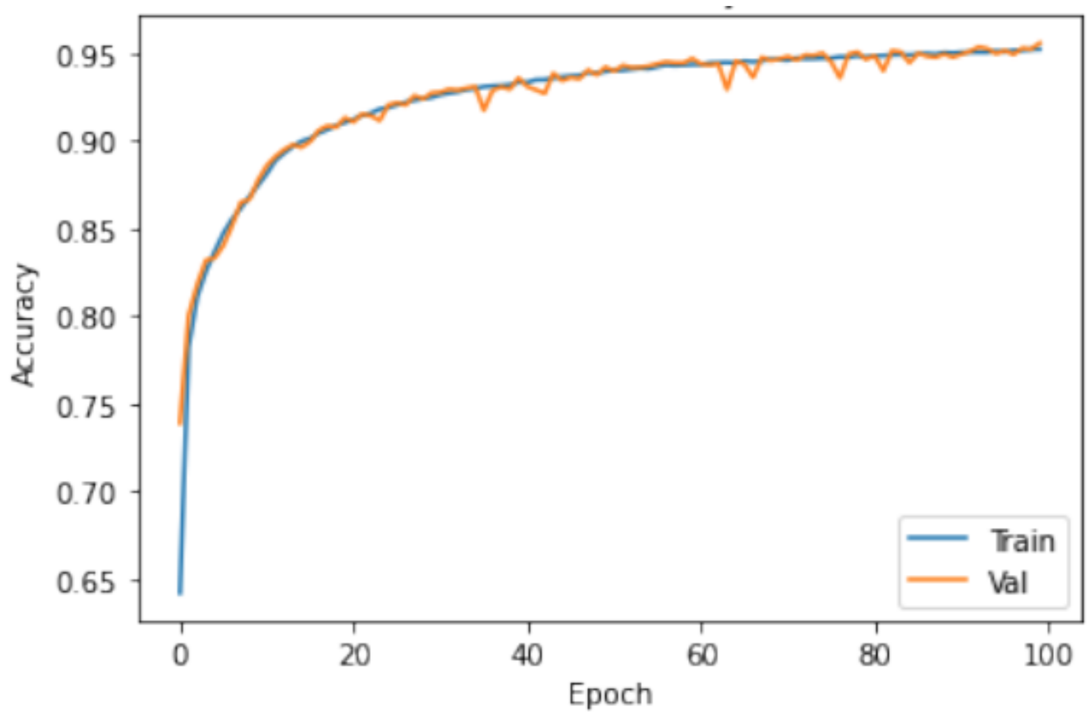

Figure 6.18: Epoch vs Accuracy using the DNN model in MFP dataset. The training dataset "Train" is shown in blue color and the validation data is shown in orange as "Val".

The test loss was reduced, and the accuracy was improved when the number of epochs is increasing (**Figure 6.19**).

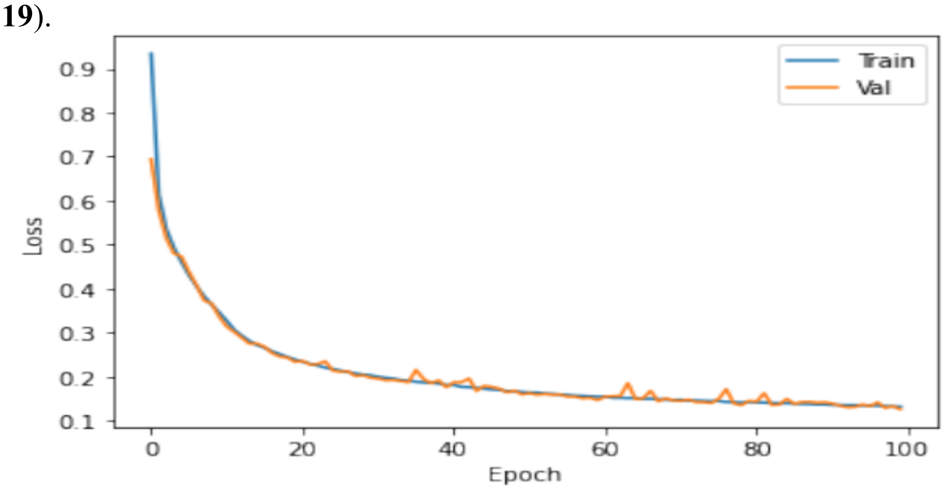

Figure 6.19: Epoch vs Loss using the DNN model in MFP dataset. The training dataset "Train" is shown in blue color and the validation data is shown in orange as "Val".

7 Discussion

7.1 Gearbox Fault Prediction

To predict the gearbox fault with ML/DNN is very challenging. As it requires a huge amount of historical data with varying equipment operating conditions to build these models. Once the data is taken from the equipment with the help of sensors and stored in the database. This data is used for training and testing the ML/DNN model. Sometimes these data required different types of preprocessing techniques before deploying the ML model. We compared the performance of our models with and without applying any preprocessing or normalization techniques on this dataset.

In the case of the gearbox, we have a binary class problem with two classes: normal and broken teeth gearbox. The data among the classes were equally distributed. So, that the classes are balanced. It is a classification problem and data are labeled. We applied supervised machine learning techniques. This dataset has five descriptive and one target feature. Seventy percent of data is used for training and thirty percent is used for testing as shown in the figure.

The performance of our ML/DLL models was not good on the raw gearbox dataset when we deployed the models on raw data without any preprocessing or normalization techniques. The accuracy, F1 score was very low. AUC score was below 0.65 and the models had difficulty differentiating between normal and broken gearbox classes. Although we have tested different ML algorithms on the raw data, the result was not much improved. The MSE error was also high. The reason for very low accuracy, an F1 score, and a very high error rate of around 40 % could be due to noise and external environment from the sensor readings. The purpose of using these techniques was to get the desired results without applying any preprocessing or normalization techniques to reduce the computational cost.

However, the performance of the ML and DNN models gradually improved when we took the sample standard deviation of sensor readings. In this case, we have used different sampling sizes such as N=10,25,50 and 100. When the sampling size has increased the accuracy, F1-score, AUC score was also increased, and the error rate was decreased. Overall, the performance of the models was significantly improved as compared to applying these models without any sampling.

This helped in removing any noise and we got desired results by gradually increasing the sample size. The accuracy and F1-score were also improved at each preprocessing and normalization step. The overall AUC score was improved to 0.98 with an average accuracy of 93%. The model easily differentiated between normal and broken gearbox classes with an AUC score of 0.98.

The algorithms were ranked based on their performance such as accuracy, error rate, F1-score, AUC score, and ROC curve. Overall performance of DNN model and random forest were very good as compared to decision tree and AdaBoost on this dataset. The ROC curves of these models became very smooth. AUC and F1 scores were also very high, and the error rates were very low. The DNN model was ranked first based on our results and the decision tree ranked last. Hence, I will suggest deploying DNN and random forest models on this type of dataset to get the desired results.

One of the drawbacks of this gearbox fault dataset was that we have classified only two types of gearbox conditions such as normal and broken teeth, but we didn't have any data and information about the other gearbox faults such as gearbox misalignment, backlash, etc.. Another limitation of this dataset was that the data taken from the simulator were recorded with predefined conditions instead of the fault occurring randomly.

7.2 Machinery Fault Prediction

Industrial machines are composed of both electrical and mechanical components. The prediction of fault in the mechanical components is challenging. Because in a single machine there are a lot of mechanical components such as gearbox, bearing, shaft, roter, etc. You will need different types of sensors to measure and observe the behavior of each mechanical component.

In the case of the machinery fault prediction dataset, the data used to build the ML and DNN model to predict the machinery fault were from the spectra quest machinery fault simulator. Unlike gearbox fault prediction, here we have multi classes such as Normal, Imbalance, Horizontal misalignment, Vertical misalignment, Underhang bearing fault, Overhang bearing fault.

The ML model is built with random forests and the performance was evaluated using the confusion matrix, accuracy, F1- score, and ROC in ML model on this dataset. The ratio of correct prediction was more than 90 percent. Accuracy, F1-score, AUC-score were also very high. MSE was very low.

Once the data is acquired from the database. The quality of data was checked, and then sample standard deviation with sampling size (N=500) was applied to the sensor's reading. This helped us to minimize any error from the sensor reading and remove any noise.

The performance of the ML and DNN models on machinery fault prediction datasets was almost the same for all the classes except the normal class. The area under the ROC curve of class normal is reduced from 0.98 to 0.95 in the DNN model. Both normal and horizontal misalignment classes are imbalanced data among other classes. So that is why their AUC scores are slightly low as compared to other classes.

One of the drawbacks of the machinery fault prediction dataset was that the data taken from the simulator were recorded with predefined conditions. The normal and horizontal misalignment classes data was very small as compared to the different faults. Although we classified different types of machinery faults in this study, in the case of bearing faults we classified only two types such as underhang and overhang faults. It would have been nice to investigate the subtype of bearing faults such as ball, rolling, and outer faults as well. This would have helped the maintenance team to know the exact type of bearing fault instead of general faults. Another limitation of this dataset was that there was a lack of data about the broken bearing and other mechanical components to build the model.

8 Conclusions

In this study performance of machine learning (ML) and deep neural network (DNN) were compared and evaluated on gearbox and machinery fault datasets. In ML we used different algorithms such as decision tree, random forest (RF), and AdaBoost to build the model. Overall, the performance of the random forest is very good as compared to the decision tree and AdaBoost.

The DNN model also performs well on both datasets, but the biggest challenges faced to build these models were the selection of hyperparameters, several hidden layers, activations functions, and loss functions to get the desired results.

Classification efficiency of ML and DNN were tested. Our results show RF and DNN models have better fault prediction ability to identify the different types of machinery and gearbox faults as compared to the decision tree and AdaBoost.

In the future, we need to investigate statistical and recurrent neural network approaches as well. Especially we need to study autoregressive integrated moving average (ARIMA) and long short-term memory (LSTM) models. The hybrid approach, which is a combination of statistical models with ML, DL, LSTM, RNN models will be very helpful in predicting missing data from the sensors.

One of the challenges of predicting faults in industrial machinery is that you require a lot of historical data to build the ML models. Industrial machines are operated in different conditions and getting the data from each component of the machine is also tricky, you require a resource to record the data from the equipment and store it in a cloud or particular place.

The biggest challenges of implementing these approaches in industries are currently IoT-based devices are only affordable for big companies and manufacturing units to monitor their equipment. We need to investigate how these ML-based predictive approaches can be transferred to small companies as well. So that they can benefit from artificial intelligence.

References

[1] “Why Implement a Predictive Maintenance Strategy?,” *Vimana*. https://govimana.com/predictive-maintenance-strategy/ (accessed Apr. 4, 2021).

[2] “Machine Learning Techniques for Smart Manufacturing: Applications and Challenges in Industry 4.0,” *Intellectyx*, Jul. 09, 2019. https://www.intellectyx.com/blog/machine-learning-for-smart-manufacturing/ (accessed Jul. 12, 2021).

[3] “What are the types of industrial maintenance and how can IIoT help?,” *Netilion Blog*, Sep. 02, 2019. https://netilion.endress.com/blog/types-industrial-maintenance-iiot/ (accessed Apr. 9, 2021).

[4] F. Ribeiro, M. Marins, S. Netto, and E. Silva, “Rotating machinery fault diagnosis using similarity-based models,” presented at the XXXV Simpósio Brasileiro de Telecomunicações e Processamento de Sinais, 2017. doi: 10.14209/sbrt.2017.133.

[5] A. Alzghoul, A. Jarndal, I. Alsyouf, A. A. Bingamil, M. A. Ali, and S. AlBaiti, “On the Usefulness of Pre-processing Methods in Rotating Machines Faults Classification using Artificial Neural Network,” *Journal of Applied and Computational Mechanics*, Jan. 2021, doi: 10.22055/jacm.2020.35354.2639.

[6] “MAFAULDA :: Machinery Fault Database [Online].” http://www02.smt.ufrj.br/~offshore/mfs/page_01.html (accessed May. 22, 2021).

[7] “Classification of faults in gearboxes — pre-processing algorithms and neural networks | SpringerLink.” https://link.springer.com/article/10.1007/BF01413861 (accessed Aug. 23, 2021).

[8] “Gearbox fault diagnosis using data fusion based on self-organizing map neural network - Zhang Qiang, Gu Jieying, Liu Junming, Tian Ying, Zhang Shilei, 2020.” https://journals.sagepub.com/doi/full/10.1177/1550147720923476 (accessed Aug. 23, 2021).

[9] “5 Types Of Industrial Maintenance Programs,” *Quality Millwright*, Jun. 19, 2019. https://www.qmillwright.com/5-types-of-industrial-maintenance-programs/ (accessed Apr. 17, 2021).

[10] M. Bentley, “Machine Learning for Predictive Maintenance - Top Opportunities for 2020-2021 | The Ritz Herald,” *https://ritzherald.com/*. https://ritzherald.com/machine-learning-for-predictive-maintenance/ (accessed May. 17, 2021).

[11] “Disadvantages of doing Preventive Maintenance,” *Bayt.com*. https://specialties.bayt.com/en/specialties/q/95601/disadvantages-of-doing-preventive-maintenance/ (accessed May. 21, 2021).

[12] “Predictive Maintenance – What you need to know | IoT Now News & Reports,” *IoT Now News - How to run an IoT enabled business*, May 02, 2018. https://www.iot-now.com/2018/05/02/81526-predictive-maintenance-need-know/ (accessed Jul. 23, 2021).

[13] “What Are Benefits and Drawbacks of Preventive Maintenance?” https://www.onupkeep.com/answers/preventive-maintenance/benefits-of-preventive-maintenance (accessed Jun. 29, 2021).

[14] “What is Predictive Maintenance? [Benefits & Examples],” *Fiix*. https://www.fiixsoftware.com/maintenance-strategies/predictive-maintenance/ (accessed Apr. 7, 2021).

[15] Y. Ageeva, “Predictive Maintenance Scheduling with AI and Decision Optimization,” *Medium*, May 15, 2020.

[16] "The importance of machine learning data | Algorithmia Blog." https://algorithmia.com/blog/the-importance-of-machine-learning-data (accessed Apr. 26, 2021).

[17] "ML | Types of Learning – Supervised Learning - GeeksforGeeks." https://www.geeksforgeeks.org/ml-types-learning-supervised-learning/ (accessed Apr. 7, 2021).

[18] "Supervised learning models – IBM Developer." https://developer.ibm.com/articles/cc-supervised-learning-models// (accessed Apr. 12, 2021).

[19] "Unsupervised Machine Learning: What is, Algorithms, Example." https://www.guru99.com/unsupervised-machine-learning.html (accessed Apr. 12, 2021).

[20] "SpectraQuest Inc.: Machinery Fault Simulators." https://spectraquest.com/products/simulators/machinery-fault-simulators/ (accessed May. 6, 2021).

[21] "Open Energy Data Initiative (OEDI)," *OpenEI*. https://data.openei.org/ (accessed Apr. 2, 2021).

[22] S. S. Shahapurkar, H. S. Pansare, P. P. Dhebe, C. S. Wagh, and A. Desale, "Detection of Fault in Gearbox System Using Vibration Analysis Method," p. 4.

[23] A. Yunusa-Kaltungo and R. Cao, "Towards Developing an Automated Faults Characterisation Framework for Rotating Machines. Part 1: Rotor-Related Faults," *Energies*, vol. 13, no. 6, Art. no. 6, Jan. 2020, doi: 10.3390/en13061394.

[24] Y. Ageeva, "Predictive Maintenance Scheduling with AI and Decision Optimization," *Medium*, May 15, 2020. https://towardsdatascience.com/predictive-maintenance-scheduling-with-ibm-data-science-experience-and-decision-optimization-25bc5f1b1b99 (accessed Aug. 11, 2021).

[25] S. Orhan, N. Aktürk, and V. Çelik, "Vibration monitoring for defect diagnosis of rolling element bearings as a predictive maintenance tool: Comprehensive case studies," *NDT & E International*, vol. 39, no. 4, pp. 293–298.

[26] "Standard Deviation | How and when to use the Sample and Population Standard Deviation - A measure of spread | Laerd Statistics." https://statistics.laerd.com/statistical-guides/measures-of-spread-standard-deviation.php (accessed Aug. 23, 2021).

[27] "Confusion Matrix for Machine Learning." https://www.analyticsvidhya.com/blog/2020/04/confusion-matrix-machine-learning/ (accessed Apr. 2, 2021).

[28] "Confusion Matrix - an overview | ScienceDirect Topics." https://www.sciencedirect.com/topics/engineering/confusion-matrix (accessed Apr. 22, 2021).

[29] S. R. Saufi, Z. A. B. Ahmad, M. S. Leong, and M. H. Lim, "Challenges and Opportunities of Deep Learning Models for Machinery Fault Detection and Diagnosis: A Review," IEEE Access, vol. 7, pp. 122644–122662, 2019, doi: 10.1109/ACCESS.2019.2938227.

[30] S. Ayvaz and K. Alpay, "Predictive maintenance system for production lines in manufacturing: A machine learning approach using IoT data in real-time," 2021, doi: 10.1016/j.eswa.2021.114598.