# Early-stage bearing fault detection using machine learning and deep learning techniques

Shashank Pandey
Department of Computer Science and Information Sciences
Birla Institute of Technology and Science, Pilani
Hyderabad, India
Email: f20200238@hyderabad.bits-pilani.ac.in

***Abstract***— Rolling bearings are among the most important components in the vast majority of machines. Their health is extremely important, as they are an integral tool in the physical functioning of a wide range of machines used in industrial sites. Apart from that, they are also some of the most vulnerable parts of a machine, consistently being under high load and continual stress. Thus, detecting bearing faults early on is of paramount importance in the real world, requiring methods to be able to make conclusions on the presence/absence of faults, and the part(s) of the machine affected by these faults. We see that the location of the bearing fault plays an essential role in detecting future faults, as we are more confident of where (and how) the fault is possibly present.

In this paper, we explore the use of a specific deep learning algorithm, using the concept of transfer learning to predict the location of different bearing faults in machines. We also compare the performance of another different learning algorithm, using the naive Bayes method on a separate dataset, intending to highlight the different approaches both algorithms take in successfully classifying such faults, and how the different datasets and methods of approach result in different levels of success.

***Keywords— bearings, deep learning, classification, faults, scalograms, transfer learning, naive Bayes***

## I. Introduction

An electric motor converts electrical energy into mechanical energy which is then supplied to different types of loads. A.C. motors operate on an A.C. supply and are classified into synchronous, single phase and 3 phase induction, and special purpose motors. 3 phase induction motors are currently the widespread industry standard, and are used by more than 70% of the global market.

A 3-phase induction motor is an electromechanical energy conversion device that converts 3-phase input electrical power into output mechanical power. It differs from a 1-phase induction motor, primarily because unlike a 1-phase induction motor, it does not require a starting capacitor. It consists of a stator and a rotor. The stator carries a 3-phase stator winding while the rotor carries a short-circuited winding, also known as rotor winding. The stator winding is supplied from a 3-phase current supply. The rotor winding drives its voltage and power from the stator winding through electromagnetic induction, which gives credence to its' name, the induction motor.

Condition monitoring is the process of monitoring a particular condition in machinery (such as vibration, temperature, etc) to identify changes that could indicate a developing fault. Continuously monitoring the condition of equipment and noting any irregularities that would typically shorten an asset's lifespan, helps us identify necessary maintenance or other preventive actions to be scheduled to address the issue(s), before they develop into more severe failures.

Bearings are parts of the motor that support the rotor, maintain a gap between the rotor and the stator, and transfer the loads from the shaft to the motor frame. They reduce friction between the various moving parts of the machines, and ensure that only the desired motion occurs. Rolling bearings, a subset of bearings on which a large amount of our work relies upon, differ from conventional bearings, as they do not rely on the principle of reducing friction between surfaces, but instead generate the required friction through the rolling motion, which is usually generated by spherical balls (as shown in Fig.1), that bears the axial load of the surface. These rolling bearings are used when higher movement loads are involved, compared to plain bearings. Bearings are present on both the drive-end and the fan end of the rotor.

*(Fig.1) An image of SKF angular contact ball bearings*

They are extremely sensitive, and important parts of the motor, and timely detection of bearing faults can prevent costly and consequential motor failure and greatly reduce losses for a manufacturing firm. Furthermore, rolling element bearings tend to wear out easily, due to the continuous metal-to-metal contact, which creates faults in three prominent locations: the outer race, the inner race, and the ball itself. It is also an extremely vulnerable component, as it is under high-load and high speed conditions. Therefore, regular maintenance and diagnostics of these rolling-element bearings is critical for industrial safety, along with reducing maintenance costs and possible shutdown time.

Existing approaches to bearing fault classification have used a series of differing methods, including artificial neural networks, classification algorithms like K-nearest neighbours, principal component analysis (PCA) and support vector machines, to mention a few However, the comparative difference of both approaches and of results between two highly different methods- transfer learning on data-based generated scalograms, and a classical Bayes' theorem-based classification approach has not yet been explored, which this paper attempts to compare and contrast. The paper also makes use of two differing sets of datasets, both widely used, to further diversify the overall scope of the problem to be answered.

## II. Datasets used

### *A. MPFT Challenge Dataset*

The first dataset used in this paper is the MPFT Challenge dataset, collected by the Society for Machinery Failure Prevention Technology. It comprised of data from a bearing test rig (nominal bearing data, an outer race fault at various loads, and an inner race fault at various loads), and three real-world faults.

The test rig was equipped with a NICE bearing with the following parameters:

- Roller diameter: rd = 0.235
- Pitch diameter: pd = 1.245
- Number of elements: ne = 8
- Contact angle: ca = 0

The dataset comprises of the following:

- 3 baseline conditions: 270 lbs of load, input shaft rate of 25 Hz, sample rate of 97,656 sps, for 6 seconds
- 3 outer race fault conditions: 270 lbs of load, input shaft rate of 25 Hz, sample rate of 97,656 sps for 6 seconds
- 7 outer race fault conditions: 25, 50, 100, 150, 200, 250 and 300 lbs of load, input shaft rate 25 Hz, sample rate of 48,828 sps for 3 seconds (bearing resonance was found be less than 20 kHz)
- 7 inner race fault conditions: 0, 50, 100, 150, 200, 250 and 300 lbs of load, input shaft rate of 25 Hz, sample rate of 48,828 sps for 3 seconds

The data contained an acceleration signal gs, sampling rate sr, shaft speed rate, load weight load, and four critical frequencies representing different fault locations: ball pass frequency outer race (BPFO), ball pass frequency inner race (BPFI), fundamental train frequency (FTF), and ball spin frequency (BSF).

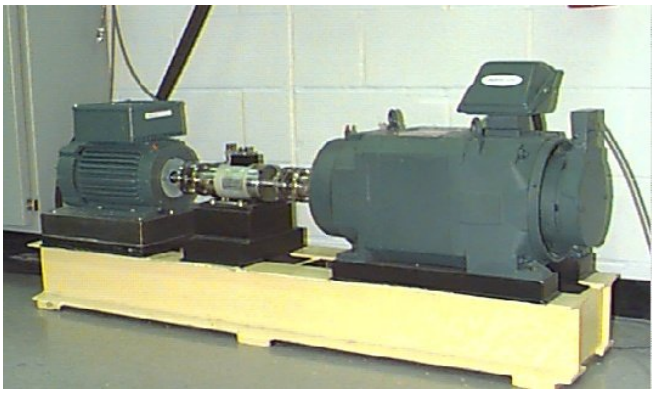
*(Fig.2) Inner race of a ball bearing in the MPFT dataset)*

The generated data was recorded and made publicly available on the MPFT website. It serves as an alternative dataset to the secondary dataset, recorded to diversify our dataset to be able to get a more accurate comparison between our two approaches.

### B. *CWRU (Case Western Reserve University) dataset*

To collect the data in the second dataset, from the Case Western Reserve University's Bearing Data Center (as expanded on below), a 2 HP, Class - 1 horizontally-mounted Reliance electric motor was used. Under the conditions of the experiment, the motor will be rotating at 1750 rpm with a load of 2 horsepower. The readings will be recorded using accelerometers with a sampling frequency of 48kHz. The bearing model will focus on is 6205-2RS JEM SKF, a deep groove ball bearing.

Single point faults are introduced to the bearings under test using electro-discharge machining with fault diameters of 7 mils, 14 mils, 21 mils, 28 mils, and 40 mils, at the inner raceway, the rolling element, and the outer raceway. Vibration data are collected for motor loads from 0 to 3 hp and motor speeds from 1,720 to 1,797 rpm using two accelerometers installed at both the drive end and fan end of the motor housing, and two sampling frequencies of 12 kHz and 48 kHz were used.



*(Fig.3) Experimental setup used for collecting the CWRU dataset*

The specifications of the drive-end bearings are as follows:

- Inner Diameter: 0.9843 inches
- Outer Diameter: 2.0472 inches
- Thickness: 0.5906 inches
- Ball Diameter: 0.3126 inches
- Pitch Diameter: 1.537 inches

The specifications of the fan-end bearings are as follows:

- Inner Diameter: 0.6693 inches
- Outer Diameter: 1.5748 inches
- Thickness: 0.4724 inches
- Ball Diameter: 0.2656 inches
- Pitch Diameter: 1.122 inches

The generated data has been recorded and been made available on the website of the Bearing Data Center. This dataset is used as it can serve as a method to validate and compare the performance of different ML and DL algorithms, between this paper and other ones as well.

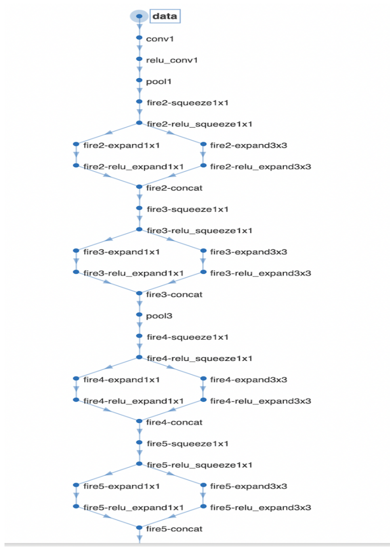
## III. METHODOLOGY AND IMPLEMENTATION

### A. *Transfer learning*

Transfer learning is the central concept behind the neural network we design. To put it formally, it is a research approach, and algorithm that focuses on storing knowledge gained while solving one problem and applying it to a different but related problem. For example, knowledge gained while learning to recognize cars could apply when trying to recognize trucks. The utilization of pre-trained neural networks helps us to modify the weights of the network to make it more generalizable, without loss of performance; and also allows us to train the network more efficiently.

The base neural network model chosen for this paper for the purposes of transfer learning, is the DAGNetwork, which is a neural network with layers arranged in the form of a directed acyclic graph. While the overarching architecture of the network might end up being very complex, the base pathway of the network is similar to that of an acyclic graph, which is a graph without any cycles (no loops) therefore, differentiating our network from a recurrent neural network.

Our specific neural network has eighteen layers. It is a convolutional network, and is used to train images. The architecture of the network is shown in figure 4. various optimization layers, pooling layers, and classification layers are part of the network.




*(Fig.4) Architecture of the neural network*

Practice school station: Electrono Solutions Bengaluru
Industry Mentor: Arun Kumar S
Faculty co-ordinator: Ashwin K.P.

*B. Naive Bayes classification*

Our other model designed works on the principle of a fairly simple probabilistic theorem.

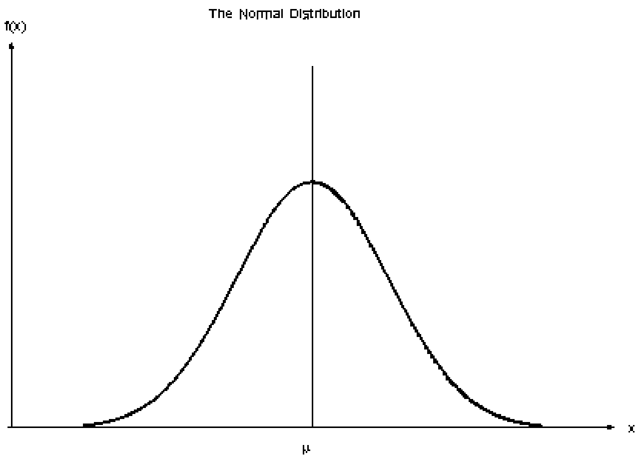
The Bayes' theorem can be expressed in the formula,

$$P(A|B) = \frac{P(A,B)}{P(B)}$$

Here, the key assumption the theorem makes is that events A and B are independent, i.e, the probability of event A occurring does not depend on the probability of event B occurring and vice versa.

Applied to a dataset, we assume that none of the features in our dataset are dependent on each other. The other central assumption we make at the outset is that all the predictors have an equal effect on the outcome, although we do give them different quantifiable levels of preference over time.

More specifically for the case of our problem, since we are dealing with acceleration data in the dataset, recorded at different distances, which is bound to be continuous in nature, we can presume that the features of the dataset would follow a Gaussian, or normal distribution, where we see a smooth curve with the highest frequency at the mean.




*(Fig.5) A Gaussian distribution*

Therefore, we can mathematically conclude that the conditional probability of a parameter ($x_i$) to be indicative, given the class variable (y) is of a specific class, is given as

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} exp\left(-\frac{(x_i-\mu_y)^2}{2\sigma_y^2}\right)$$

where σ refers to the standard deviation of the variable, and μ is the population mean.

Ultimately, we get the conditional probability of a data-point being classified as class $c_j$ as,

$$\frac{\prod_i \frac{1}{\sqrt{2\pi\sigma_{i,j}^2}} e^{-\frac{1}{2}\left(\frac{x_i-\mu_{i,j}}{\sigma_{i,j}}\right)^2} \cdot p(c_j)}{p(x)}$$

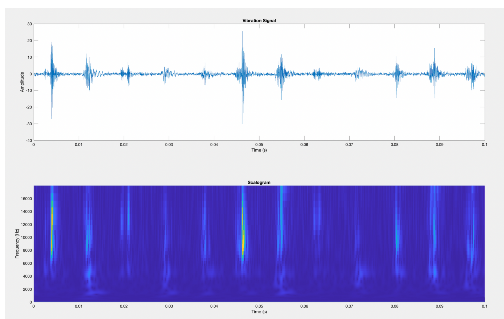
Among all classes, the class for which the numerator here has the highest value is the class to which we classify the specific data point.

*C. Implementation*

To implement our transfer learning-based model, MATLAB is used. The MPFT Challenge dataset, which consists of files, having vibration signals of each of the features, is used to train the dataset.

However, since we are implementing an image-based neural network, it is essential to convert our raw data into some form of an image, such that we can use it to train the network. For this purpose, in the first part of our program, we convert the vibration signals to scalograms. Scalograms are two-dimensional (frequency, time) representations of the original time-domain vibrational signal that we have in the dataset. They are used as they are representative of the different amplitudes of vibrations of all the three classes in the dataset (inner race fault, outer race fault, and normal case). The conversion is done using envelope analysis of the signal values, a technique of amplitude demodulation, which allows us to remove noise and obtain the original waveform.

The generated signals are now used as our data for training the model. We split the training and validation datasets, along with the test dataset. Following this, we train the dataset, using the modified DAGNetwork as mentioned above, with stochastic gradient descent used as the loss function, and record the results.



*(Fig.6) One of the generated scalograms, with the original bearing signal above.*

To implement the second model, the naive Bayes-based classifier, we use Python. We initially run a script that converts the raw data across different files, into comma-separated values.

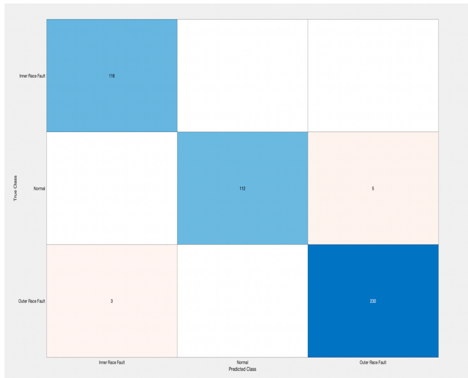
We note here that while the first dataset we used (the MPFT dataset), for transfer learning, had only three class labels (inner race fault, outer race fault, and normal), the CWRU dataset has ten class labels (it comprises of ball faults as well, and all three types of faults have three different classes, depending on the fault diameter. Therefore, when we perform comparative analysis, we have to modify our dataset, to merge the different classes of inner and outer race faults into one class each, and remove the ball fault class altogether, as that cannot be replicated in the MPFT dataset.

Since we are attempting to fit a Gaussian distribution on our predictor variables, we then normalize all the values using a scaler to have null mean and unit variance. We split the data into test and training sets, for cross-validation. Following which, we use Python's scikit library to create a naive Bayes classifier, and define its' parameters. We then fit the model on the training set, and use it to predict the class values of the test set. We, finally, find metrics of our model's performance, and generate a confusion matrix.

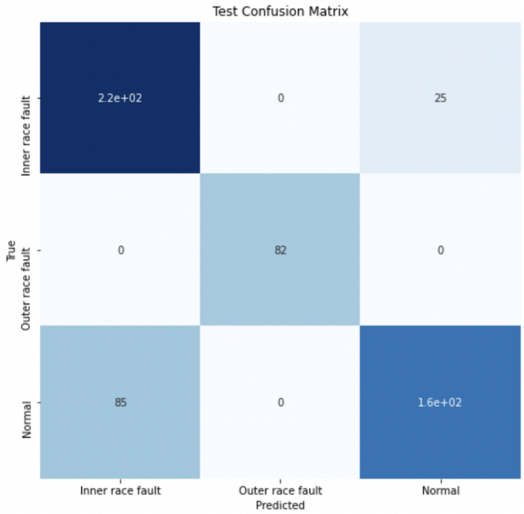
## IV. Results

We find that the transfer-learning model, with an accuracy of 99.42%, greatly outperformed the naive Bayes classifier, which showed an accuracy of 80.87%. From the confusion matrices of both models (see Fig. 7 and 8), we can see that while most classes were easily handled and correctly predicted by both models, the naive Bayes classifier made a large number of misclassifications between cases where there was no fault, and where there was an inner race fault. These cases accounted for all the misclassifications in the model, with no misclassifications where we had an outer race fault. This leads us to assume that the central, 'naive' assumption of the naive Bayes classifier (of conditional independence between predictors was not true.)

The transfer learning-based model, on the other hand, only showed very few errors, misclassifying some outer race faults as inner race faults and some normal cases as outer race faults. However, its overall incredible performance speaks to the incredible potential that transfer learning-based approaches have.



*(Fig.7) The confusion matrix of the transfer learning-based neural network result.*




*(Fig.8) The confusion matrix of the naive Bayes classifier-based result. We see a significant number of errors, with no fault cases incorrectly being predicted as having an inner race fault.*

Another interesting thing to note is that when asked to perform classification between the original ten classes of the CWRU dataset, rather than our modified and simplified 3-class version, we see that the naive Bayes classifier has an accuracy of almost 89%, a significant improvement over its performance in our simplified dataset. This highlights that the naive Bayes classifier, is conversely quite strong at making predictions regarding the size of bearing faults, but not that powerful when trying to predict the location.

Future work in this region may be done by attempting to fit more conventional neural networks on the raw dataset, and compare it with being fitted to the generated scalograms. Other classification techniques like random forests could also be trialed in place of the naive Bayes approach used here.

## Acknowledgment

I would like to express my gratitude to Professor Ashwin KP, our faculty mentor from the mechanical engineering department of BITS Goa, and Mr. Arun Kumar, Research Director at Electrono Solutions Pvt Ltd for their contributions to this project. Their endless support, encouragement and driving spirit has been essential to the work done in this project.

I would also extend my thanks to to Mr. Khadar, of BITS Pilani Hyderabad Campus, for providing LabVIEW software access and to Ms. Nikitha for demonstrating dataset integration into LabVIEW.

Without the help and support of the abovementioned individuals, this project would not have been possible.

## References

[1] Zhang, Shen & Zhang, Shibo & Wang, Bingnan & Habetler, Thomas. (2019). "Machine Learning and Deep Learning Algorithms for Bearing Fault Diagnostics - A Comprehensive Review"

[2] Sadoughi, Mohammakazem, Austin Downey, Garrett Bunge, Aditya Ranawat, Chao Hu, and Simon Laflamme, "A Deep Learning-based Approach for Fault Diagnosis of Roller Element Bearings," 2018 Annual Conference of the Prognostics and Health Management Society, Proceedings of the Annual Conference of the PHM Society 10, no. 1 (2018). DOI: 10.1234/phmconf.2018.v10i1.526.

[3] R. Magar, L. Ghule, J. Li, Y. Zhao and A. B. Farimani, "FaultNet: A Deep Convolutional Neural Network for Bearing Fault Classification," in *IEEE Access*, vol. 9, pp. 25189-25199, 2021, doi: 10.1109/ACCESS.2021.3056944.

[4] Neupane, D., & Seok, J. (2020). Bearing Fault Detection and Diagnosis Using Case Western Reserve University Dataset With Deep Learning Approaches: A Review. IEEE Access, 8, 93155–93178. https://doi.org/10.1109/access.2020.2990528.

[5] Zilong, Zhuang & Qin, Wei. (2018). Intelligent fault diagnosis of rolling bearing using one-dimensional Multi-Scale Deep Convolutional Neural Network based health state classification. 10.1109/ICNSC.2018.8361296.

[6] Case Western Reserve University, Case School of Engineering, link: https://engineering.case.edu/bearingdatacenter/download-data-file

[7] The MPFT Challenge Dataset, Society for Machinery Failure Prevention Technology, link: https://www.mfpt.org/fault-data-sets/

[8] N. Sawalhi & R.B. Randall (2017), "Simulating gear and bearing interactions in the presence of faults Part I. The combined gear bearing dynamic model and the simulation of localised bearing faults"

[9] David Verstraete, Andrés Ferrada, Enrique López Droguett, Viviana Meruane, Mohammad Modarres, "Deep Learning Enabled Fault Diagnosis Using Time-Frequency Image Analysis of Rolling Element Bearings", Shock and Vibration, vol. 2017, Article ID 5067651, 17 pages, 2017. https://doi.org/10.1155/2017/5067651

[10] Li, Bo & Chow, Martina & Tipsuwan, Yodyium & Hung, J.C.. (2000). Neural-network-based motor rolling bearing fault diagnosis. Industrial Electronics, IEEE Transactions on. 47. 1060 - 1069. 10.1109/41.873214.

[11] Bechhoefer, Eric. "Condition Based Maintenance Fault Database for Testing Diagnostics and Prognostic Algorithms." 2013

Practice school station: Electrono Solutions Bengaluru
Industry Mentor: Arun Kumar S
Faculty co-ordinator: Ashwin K.P.