



[Купить книгу "New Trends in Electric Machines"](#)

IntechOpen

New Trends in Electric Machines

Technology and Applications

*Edited by Miguel Delgado-Prieto,
José A. Antonino Daviu and Roque A. Osornio Rios*



Fault Detection and Predictive Maintenance of Electrical Machines

*Hadi Ashraf Raja, Karolina Kudelina, Bilal Asad
and Toomas Vaimann*

Abstract

Nowadays, most domestic and industrial fields are moving toward Industry 4.0 standards and integration with information technology. To decrease shutdown costs and minimize downtime, manufacturers switch their production to predictive maintenance. Algorithms based on machine learning can be used to make predictions and detect timely potential faults in modern energy systems. For this, trained models with the usage of data analysis, cloud, and edge computing are implemented. The main challenge is the amount and quality of the data used for model training. This chapter discusses a specific version of a condition monitoring system, including maintenance approaches and machine learning algorithms and their general application issues.

Keywords: electrical machines, fault diagnostics, predictive maintenance, artificial intelligence, condition monitoring, neural networks

1. Introduction

The recent advancement in information technology, especially the integration of technology with different fields of research, has made day-to-day life convenient and opened up new research areas. One of these fields is the internet of things (IoT), which enables physical devices to communicate through the internet. The advent of these smart or intelligent devices and their implementation in industrial applications resulted in the industrial revolution, commonly known as industrial standard 4.0. These devices are not only able to communicate with each other but also able to make decisions based on defined logic or controlled remotely also referred to as cyber-physical systems. This has further paved the way for condition monitoring of electrical devices, where these devices act as data acquisition points. The collected data can then be used to monitor specific electrical machines. Further, data analysis can be done on the collected data to include fault diagnostics on these devices, including the prediction of faults [1, 2].

Industrial standard 4.0 has given way to the implementation of condition monitoring [3, 4] at a mass scale in the industry, leading toward predictive maintenance [5, 6] of electrical machines in the near future. Many companies are working on different predictive maintenance algorithms to reduce their scheduled maintenance costs. This research will further improve the effectiveness of electrical machines in the industry [7] and help to reduce unforeseen errors and faults. Most companies are also researching and finding the lifespan of the equipment based on previous patterns and external environmental variables to get the best results out of their setup. Researchers have already implemented different condition monitoring setups to maximize the potential of different electrical machines, including offshore wind turbines [8, 9], but most of this equipment is expensive and heavy.

At the moment, the industry is trying to move toward predictive and proactive maintenance to help reduce costs due to unexpected errors and faults that could have been handled before they become a more significant issue. The maintenance of electrical machines is usually divided into four phases: reactive, periodic, proactive, and predictive, as shown in **Figure 1**. Among the four phases, most of the industry is still on scheduled maintenance but is trying to move toward predictive maintenance as it is not only cost efficient but also generates a more detailed report on fault diagnostics.

With the move toward predictive maintenance, researchers are also looking for ways to utilize newer technology to get better results. The research is not only going on in this area but also in other areas like wearable devices for condition monitoring of patients to check on any abnormality [10, 11], solar-powered condition monitoring systems [12], air and noise pollution monitoring systems [13], and much more. This is because of the advancement in the technology of microcontroller boards that have given researchers more options to explore. More researchers are including these boards in their research because of their scalability. There have already been researches going on like the development of a condition monitoring system for wind turbines [14], weather sensors [15], electrical machines [16–18], autonomous vehicles [19], and robotics [20, 21]. Most of these condition monitoring systems are still in development and might need much more improvement before they can become stable and be used on a large scale. One of the most common issues is the sample rate at which data is gathered using these devices and its transmission without any data loss.

Most of the systems already in place use SCADA/PLC that are not only complex and expensive but also harder to transport [22]. One of the other issues with these systems is that although they are data acquisition points, there is no data analysis of the collected data. Hence, it is just lying there and not being utilized anywhere for fault diagnostics or being used to deduce any results. For the analysis of the collected

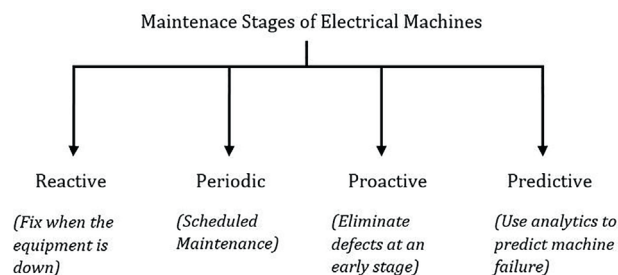


Figure 1.
Maintenance of electrical machines.

data, cloud computation is used along with edge computing, which helps analyze the data and deduce results from it. For the analysis part, machine learning algorithms are mainly used to train models based on collected data from these machines. These trained models are implemented on the cloud to get near accurate classification and prediction related to incoming data from the electrical machines. These models mainly were implemented on cloud storage or isolated servers as they need high processing power and storage space. However, now things are moving toward edge computation from the cloud. This will result in these models being implemented at the edge node where the data is being collected rather than on the cloud, which will help identify errors on the edge and further reduce the time needed to make a decision. This will also result in reduced bandwidth needed to transfer the data over the network.

This chapter discusses a concise overview of a condition monitoring system using microcontroller cards, following a small data pre-processing and analysis. Further, some light is shown on the machine learning algorithms and the training of data sets for different faults, and a short detail related to predictive maintenance is given, how it can help, and at what stage it is currently at, followed by a short conclusion.

2. Condition monitoring system

This section will discuss a particular approach to condition monitoring systems based on microcontroller boards and cloud resources. The condition monitoring system technically consists of three parts: the data acquisition system, the edge node, and the cloud. Usually, the researchers do not consider an edge node system. However, it is always better to have a local backup, computation power, and space to run some analysis if needed. The data acquisition part will consist of the microcontroller board, with the edge node being the one that helps in case of any data loss over the network.

The data acquisition part will gather data from the electrical machine using sensors. The incoming data is calibrated before transferring it through the microcontroller board to the edge node. In most cases, as the industry uses analog sensors, this part also acts as an analog-to-digital converter (ADC). The acquired data is then transmitted toward the second part that is, the edge node. The edge node acts as a local backup where the incoming data is stored in a MySQL database. The database is synced in real time with the database present in the cloud. Some pre-processing can also be done on edge, including digital filtering. The third part of the system, which is in the cloud, runs the frontend UI for the end user. It also runs diagnostics in the background on the latest synced data to look for faults. As the time difference between data acquisition from the electrical machine to showing the diagnostic results on the front end is not much, this system can also be referred to as a real-time condition monitoring system. **Figure 2** shows a rough flow chart of the implementation of a condition monitoring part.

The data is collected from the electrical machines using a microcontroller, an Arduino, or a teensy. One of the microcontroller cards (i.e., Arduino) is shown in **Figure 3**. The collected data is read through one of the analogs or digital pins on the microcontroller card, depending on the sensor used for data collection. If the sensor is IoT compatible, the data can be read over on the digital pin. In contrast, the general analog sensors used in the industry need to be calibrated and their output adjusted before they can be passed onto the microcontroller board. As the pins on the microcontroller boards do not allow a negative voltage or more than a specific voltage, before providing the data to the pins of the board, it is necessary to make sure that the sensor output is calibrated correctly. If by any chance, there is a negative voltage or

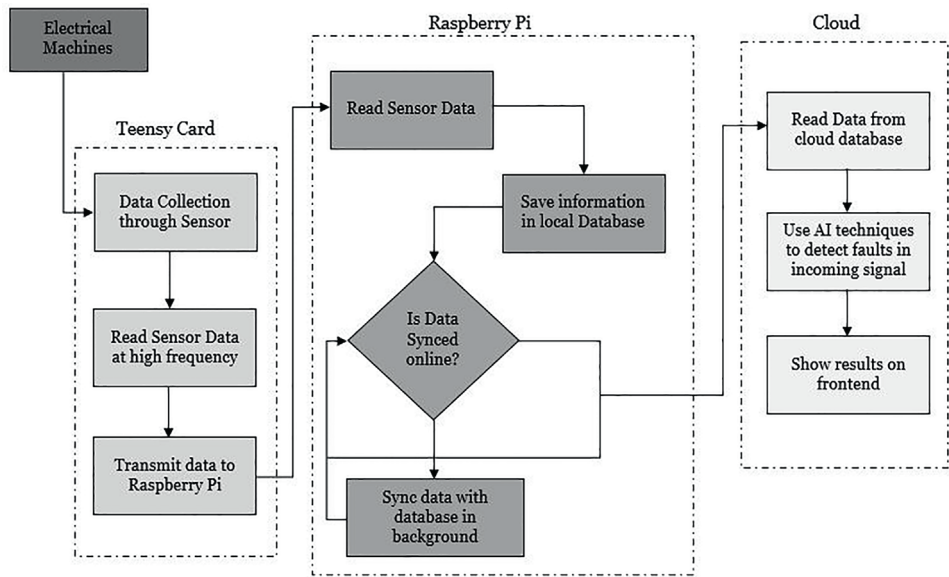


Figure 2.
Flow chart of a condition monitoring system.

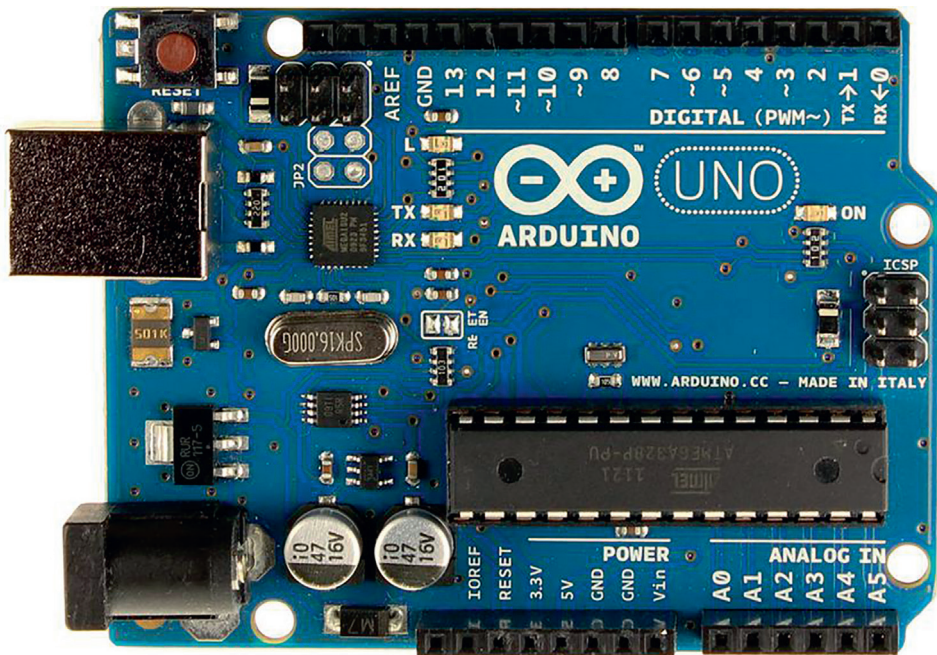


Figure 3.
Arduino board.

higher voltage than the one pin can handle, there is a high chance that the board will short circuit. So, it is essential to make sure that this is handled correctly; otherwise, might end up in a short circuit of the board and with the data collected being junk without any real meaning.

Communication method	Sample rate per second
UART	1800
I2C	2600
SPI	3600

Table 1.
 Comparison of sample rate for different communication methods.

Once the data is received on the board, it is then forwarded to the edge node, which is made up of Raspberry Pi. The data read here through the analog pin is at high speed. To ensure, it is transmitted at the same speed without any loss of data serial peripheral interface (SPI) connection is used between the microcontroller board and Raspberry Pi. Also, to be sure, the voltages for both the microcontroller board and Pi are different as some microcontroller boards give an output of 5 Volts at high. In contrast, Pi works with a voltage of 3.3 Volts when high, so it is also needed to ensure the transmitted values do not go over it. If a high sample rate is not needed, then UART communication should be preferred. A short description of different communication methods and their sample rate for a longer period of time is shown in **Table 1**.

The above sample rate per second is just a comparison between the speed for different communication methods for a specific microcontroller board. In this case, the microcontroller board is considered as Arduino Mega. The communication devices and other specifications, including the buffer capacity of the logger device, are the same in all three cases, that is ., Arduino Mega and Raspberry Pi. The results shown in **Table 1** are approximately the maximum sampling rates of an Arduino Mega that can be achieved when run for over a couple of days with the specific communication method without any data loss during transmission from Arduino Mega to Raspberry Pi. These specific results are hardware-dependent and changing the microcontroller board will change the speed range, example, teensy has a far better range. An experimental setup with an induction motor and analog current sensors for data acquisition is shown in **Figure 4**.

The communication method for data transmission between the microcontroller board and Pi can be decided based on the sample rate needed for the transmission. These sample rates are based on continuous data transmission from a couple of hours to days without any data loss between the transmissions. Similarly, the choice of microcontroller board might also impact the sample rate for transmission, as the newer board having better computation power gives better results. Once the data is transmitted to Pi, it is saved up in a local database and synced online simultaneously to ensure that every bit of data is synced online with the cloud without any loss. Pi also acts as a node that is capable of running analysis (like digital filtering) if needed going forward. The transmitted data is then analyzed on the cloud and based on different trained models; results are deduced whether any fault is present or not. As it is harder to understand incoming data in numerical form, the deduced results are then shown at a front end hosted on the cloud. The graphical user interface (GUI) is user-friendly and helps the end user understand the result without much information related to the system. An example of such a GUI is shown in **Figure 5**.

The GUI shown in **Figure 5** runs on the cloud with scalable resources. It mainly consists of two parts, the GUI and the diagnostic analysis running in the background. The GUI is built using PHP, whereas the diagnostic analysis primarily uses Python as the primary language, with the results saved in a MySQL database. The saved results

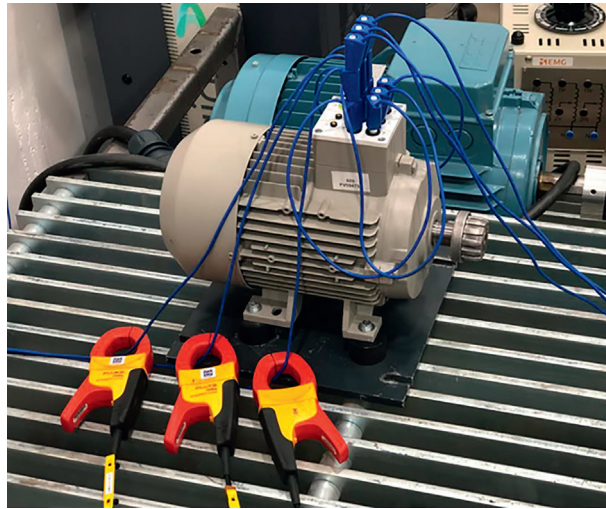


Figure 4.
Experimental setup of induction motor with analog sensors.

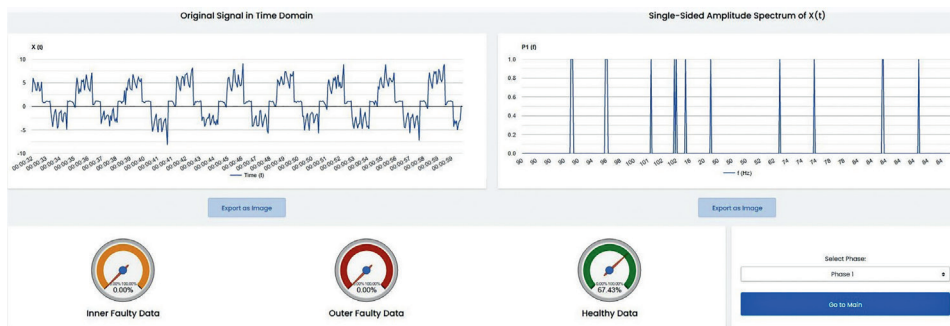


Figure 5.
Example of GUI.

are then projected on the GUI as soon as they are updated in the database. The cloud resources used here are scalable. With low data processing, that is, only one or a couple of diagnostic analyses running in the background, resources with 4 vCPU cores, 16 GB RAM, and 128 GB disk are good enough. This can be further scaled up depending on the number of diagnostic analyses and edge devices connected with the cloud, that is, increased incoming data flow.

Further, analysis results can also be shown on the GUI including the chance of a fault occurring in each phase and the option to control the electrical machine remotely if power to the machine is routed through the microcontroller board. Hence, there are multiple ways this system can be extended further. This can help the end user to understand the situation of the electrical machine in more detail. This can also help to identify which phase of the electrical machine or which part of the machine is generating issues, which can further reduce the time taken to identify the root cause of the fault. This helps maintenance teams in reducing the time needed to fix it and decide whether the fault needs to be fixed urgently or can be done later.

3. Data pre-processing and analysis

The incoming data needs to be pre-processed before it can be used for analysis. In this chapter, the analyses are focused on the steady-state operation. As the data is coming in the time domain and is raw, it is needed to make sure whether it can be utilized for the need or not. To detect faults in the early stage, it is reasonable to consider small frequency components by taking Fourier transforms of the incoming signal. For effective fault detection, different operating conditions must be considered, such as control environment, load, ambient environment, etc. **Figure 6** presents the current frequency spectra of a motor with broken rotor bars in several control modes—grid fed, scalar, and directtorque control. As seen, a significant shifting in frequency components occurs between the signals in different control modes. This is important to be considered during the model training.

At the same time, load also should be considered. **Figure 7** presents the current frequency spectra of a motor with broken rotor bars under different loads. It is seen that the behavior of the signal changes as the load increases.

In both cases, there are two regions to be studied to make predictions. Firstly, the frequency range of 0–500 Hz, where the impact of the fault is the highest on even harmonics. Specifically, the most prominent are harmonics on 50, 250, and 350 Hz. Besides, harmonics at 750 Hz can be important to be studied and considered for fault prediction.

The data is first converted into the time domain and sampled, according to the sampling frequency to make sure we have enough cycles. **Figure 8** shows an example of sample data set in the frequency domain. As the time domain does not have significant components based on which healthy and faulty data can be distinguished; hence, the data is converted into the frequency domain first and the frequency spectrum is analyzed to find the specific difference between the healthy and the faulty electrical machines.

The frequency spectrum of a faulty electrical machine includes different frequency components, usually not present in a healthy electrical machine frequency spectrum. Identifying those components and utilizing different analyzing techniques to identify them in the incoming data is part of fault detection. Including those

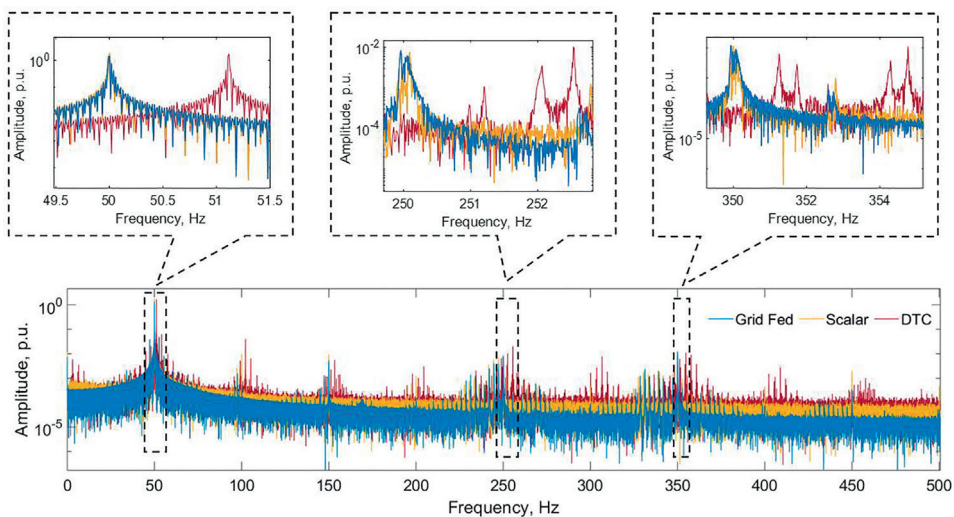


Figure 6.
 Current frequency spectra of a faulty motor under different control modes.

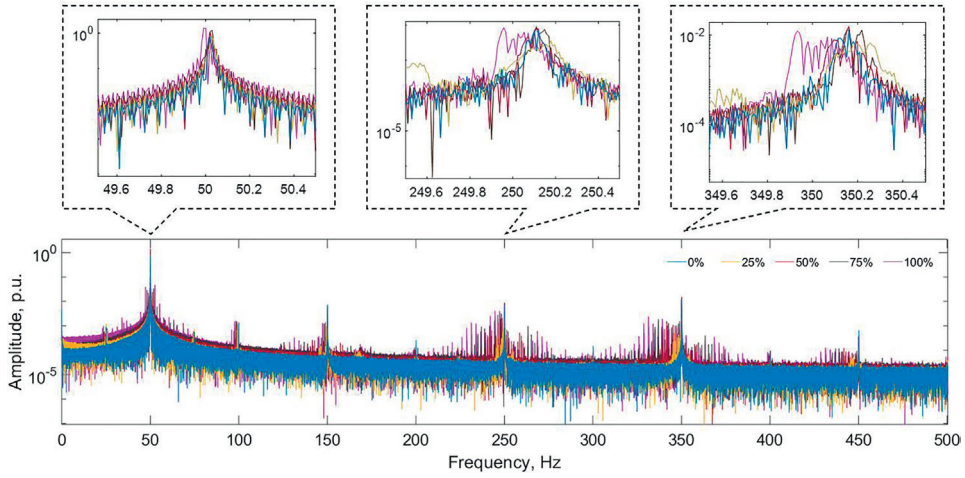


Figure 7.
Current frequency spectra of a faulty motor under different loads (grid fed).

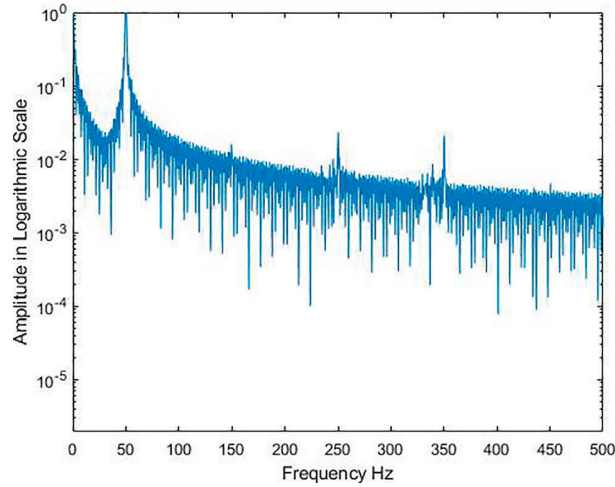


Figure 8.
The frequency spectrum of the signal to be trained.

frequency components to extract as features for training different machine learning, models can help to identify electrical machines' faults. Fault detection can be divided into two parts: signal processing and machine learning trained algorithms. Different analyses based on fast Fourier transforms can be used for the signal processing part.

4. Machine learning algorithms

The most common technique used for the detection of faults at the moment is utilizing machine learning trained algorithms. With the advent of artificial intelligence, making self-learning or systems with the aptitude for the decision has helped streamline multiple processes. Machine learning algorithms help to create a complex

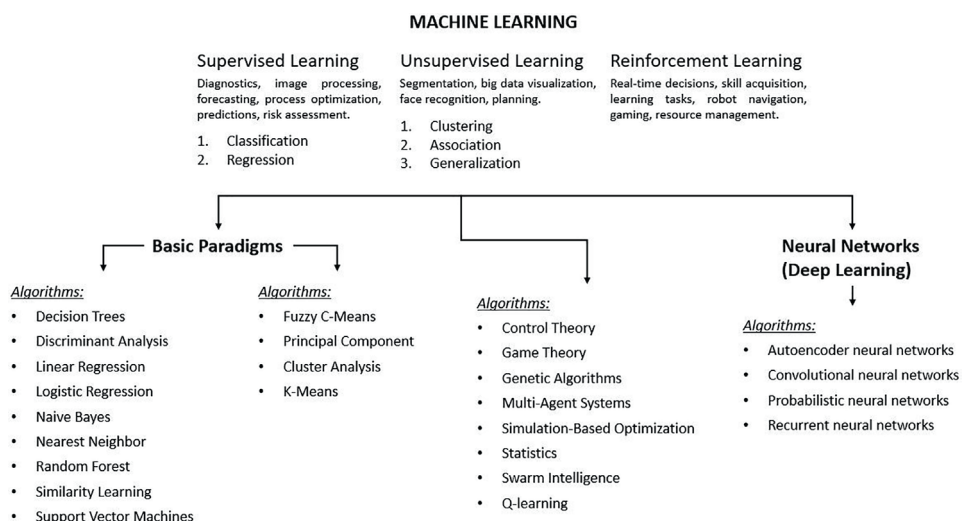


Figure 9.
Examples of machine learning algorithms [23].

weighted combination based on training data that can be used later to deduce results for the incoming data. **Figure 9** presents examples of the mostly spread machine learning algorithms in the diagnosis of electrical machines [23].

One of the primary drawbacks of machine learning algorithms is that they need a lot of data to train a high-accuracy model. However, it usually depends on the complexity of the model. Suppose the model will be used for classification, with classification being divided into two labels. In that case, the accuracy will be pretty high even with a low training data set. But suppose that is to be changed by classifying the classification into four different labels. In that case, the system's complexity will increase, resulting in the algorithm needing more data to make an accurate model. **Figure 10** shows the general working of a machine learning or neural network model, to be precise.

There are different types of machine learning algorithms based on specific logic. The training data set results in a statistical complex function based on the selected algorithm that gives a trained model. Among the machine learning algorithms, the most used are neural networks. Neural networks are further divided into three main types:

- artificial neural network (ANN),
- recurrent neural network (RNN),
- convolution neural network (CNN).

ANN and RNN are primarily used for training for models related to detection or prediction. Most ANN models are regression-based or feed-forward models, whereas RNN is feed backward neural network models. Neural network model training is divided into three layers: the input layer, the hidden layer, and the output layer. The hidden layer is where the weighted nodes are set up, as the weight of these nodes is adjusted with each training data set. Once the model is trained using the training data set, a blind validation can be carried out to test the accuracy of the model before implementing it in a real-time scenario.

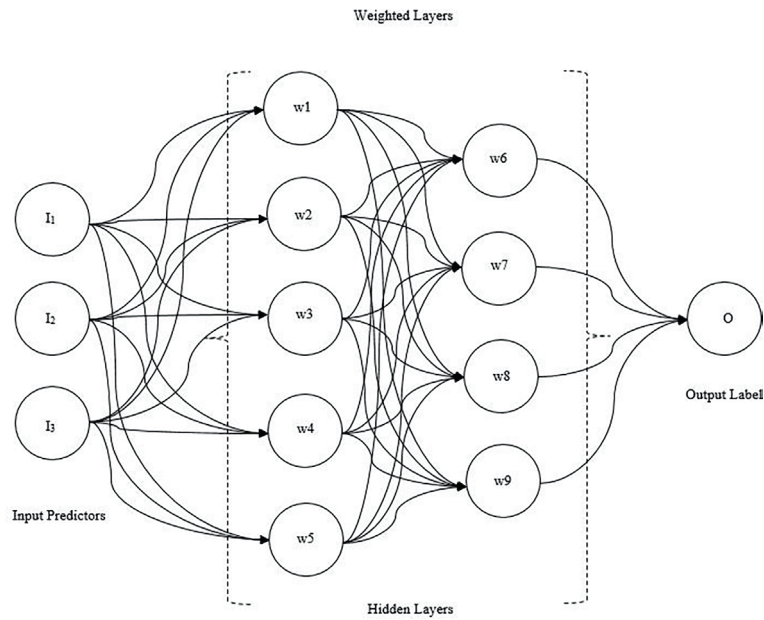


Figure 10.
Neural network schematics.

These models, after training, are usually implemented in the cloud and are used to detect faults in the incoming signals. Although they can be trained to be precise, the data needed for it is usually great. That is why researchers are looking into generating such data programmatically based on the real-time collected data and frequency harmonics. If this is reached, it will be possible to mass produce faulty data according to the need of the electrical machine. This will help train a machine learning model specifically for the required scenario. Implementing these models on nodes or edge devices might also be possible. This will help move from cloud computation toward edge computing.

Training of machine learning models also has other issues with accuracy, based on the complexity of the system. **Table 2** shows the comparison of accuracies for two different types of training sets having different labeled categories. The comparison

Neural network algorithm	Smaller training set		Bigger training set	
	Two categories (%)	Four categories (%)	Two categories (%)	Four categories (%)
Narrow Neural Network	88.30	65.00	70.00	38.50
Medium Neural Network	82.50	63.30	73.50	46.50
Wide Neural Network	88.30	73.30	76.20	51.50
Bilayered Neural Network	82.10	62.10	75.20	43.10
Trilayered Neural Network	95.40	64.20	73.3	53.40

Table 2.
Accuracy comparison of different neural network models.

confirms that changes in complexity or size of training sets do impact the accuracy of different machine learning models. In this specific example, the algorithms were run with specific conditions to compare them under similar training and validation processes. However, the results can still be optimized as the training process (i.e., epochs, etc) and the test approach (i.e., v-fold cross-validation, holdout validation, etc) can also result in different results. Hence, changing approaches can result in better or, even in some cases, worst performance, for example, a trilayered neural network with two categories and a smaller training set can result in an overfitted model.

As the system becomes more complex, a larger number of data is needed, but this also shows that there is a chance that another machine learning algorithm can perform better for the same scenario. Hence, these trained models are still flexible and there is a need to either get the optimal number of data sets for the training of the machine learning-based models or implement a custom-made machine learning model that can help identify faults related explicitly to electrical machines with high accuracy.

5. Predictive maintenance

As the industry is moving toward predictive maintenance from scheduled maintenance, there is still much research to be done in this area. Most of the research going on is related to fault detection rather than fault prediction, but companies are working in this specific area. The most important thing in this field is to identify the faulty frequency components in the early stage of the fault and the behavior of the signal and its frequency components when the signal is shifting from a healthy state to a faulty state. Once these things are identified, the next step is to train such a model that will be able to predict whether the fault is going to occur and in how much time. This will depend based on pre-processing of data and classification of the components. This is not a small task and needs dedication and time.

Researchers are looking for better ways to get a prediction model for faults help to identify them even before they occur. This leads us toward predictive maintenance, there might be some companies that are already running some kind of predictive maintenance algorithms with their systems, but at the moment, the hardware setup they have to use alongside it is quite expensive. So, another main issue in this area is to make it such that it is not only cheap but also no specific hardware setup is needed in this regard. There are also multiple directions in which predictive maintenance trained algorithms can be utilized. There can be a combination of different algorithms to get higher accuracy or more accurate results. Similarly, fuzzy logic systems can also be used in accordance with machine learning algorithms and signal processing to get a more accurate system for predictive maintenance.

Another issue that the researchers commonly face in this aspect is the lack of data. As the data collected in an industrial environment is limited, especially in the case of faults, training a model with quality data and testing it out is quite difficult. Also, the data required to train models properly should be good in quality and quantity. Some researchers are working on observing the pattern in different faults to generate a statistical equation for the faults so that synthetic signals can be generated, which can help cover up this issue. The main issue in this aspect is to correctly identify the range of amplitude of frequency components that are generated when a fault is present in the electrical machine. This is not easy as it requires much data analysis and robust testing, but immense research is taking place in this direction.