Chapter

IoT-Based Decision Support System for Health Monitoring of Induction Motors

A. Pravin Renold and B. Venkatalakshmi

Abstract

An electrical motor is a common device that is used for a variety of electrical purposes. Because of their wide range of applications, motors that are both reliable and long-lasting are in high demand. Motors are prone to a variety of faults, including rotor bar breaking faults, short turn faults, bearing outrace faults, and so on. Unexpected faults or failures in these motors reduce workplace productivity. The time it takes to resolve the issues reduces the organization's profit. Bearing failures account for approximately 42% of all faults. Due to continuous operation, the shape of the majority of electrical motors with rolling bearings becomes disproportional. This causes the motor's elastic limit to be exceeded, as well as fractures, vibrations, and a rise in temperature. A good solution is to switch from scheduled maintenance to predictive maintenance, which is based on monitoring the motor's operating condition. This chapter proposes an Internet of Things (IoT)-based solution that continuously monitors and records the vibration from the induction motor. A decision support system analyzes the impact of vibration using log data and the Naïve Bayes classifier. The proposed decision support system detects the critical level of vibration and notifies the user of the motor's abnormal working condition.

Keywords: health management, induction motor, mitigation measures, decision support system, Naïve Bayes classifier

1. Introduction

In this third era of computing, the access of predictive information from remote locations becomes essential for optimal business solutions. This is applicable for every field of applications such as agriculture, manufacturing, healthcare, education, etc. The thumb rule for the achievement of the above information access is the development of an interface between recent technologies along with proven techniques of prediction and communication. This chapter presents one such effort of interfacing Internet of Things (IoT) technology along with conventional sensing and prediction techniques. The potential solution presented in the chapter can find its role in all possible fields, which uses induction motors.

The chapter framework is composed of four major sections. The following section is an introductory part highlighting the fundamentals of various induction motor faults and the basics of IoT technology. Section 2 of the chapter describes the problem statement along with the solution using the IoT architecture. The predictive maintenance designed as a decision support system using Naïve Bayes classifier algorithm has been elaborated in Section 3. Section 4 of the chapter highlights the chapter's outcomes and the performance metrics for the proposed IoT-based solution.

1.1 Fundamentals

The chapter deals with optimal solution of induction motors and smart monitoring and precaution effects of induction motors. The key fundamentals required for such architecture are the induction motor fundamentals and the Internet of Things fundamentals. So in this section, let us review the basics of them.

1.1.1 Construction of single-phase induction motor

The single-phase induction motor has a stationary part called stator and the rotating part called rotor. The stator made up of stampings carries the winding called stator winding. It is excited by a single-phase AC supply. The number of poles (P) for which stator winding wound decides the synchronous speed of the motor. The synchronous speed is given by Ns.

$$Ns = 120f/P r.p.m. \tag{1}$$

where f is the frequency. The induction motor always rotates at a speed that is slightly less than the synchronous speed. The rotor is a rotating part of induction motor. The rotor is connected to the mechanical load through the shaft. The rotor of the three-phase induction motors is further classified as:

i. Squirrel cage rotor

ii. Slip ring rotor or wound rotor or phase-wound rotor

Depending upon the type of rotor used, the three-phase induction motor are classified as:

i. Squirrel cage induction motor

ii. Slip ring induction motor or wound induction motor or phase-wound induction motor.

1.1.2 Working principle of induction motor and different types of faults

When AC supply is given to stator winding, it carries an alternating current, which produces the alternating flux. This flux links with the rotor conductors and due to mutual induction, the rotor experiences induced e.m.f. The rotor current produces another flux called as rotor flux, which is required for motoring action.

Like every machine, the electrical motors are also prone to various faults under different operating factors and lifetime. The commonly arising problems in electrical motors are [1] as follows:

i. Stator faults

ii. Broken rotor bar or cracked rotor end-rings

iii. Air-gap irregularities

iv. Shorted rotor field winding and

v. Bearing failures

Most of the electrical motors use rolling bearing, which is used for the smooth rotational movement of the rotor. A bearing consists of two rings, one inner and the other outer. A set of balls placed in raceways rotates inside these rings [2]. The bearings are affected by the stress caused by vibration, eccentricity, and bearing currents [3]. Around 40–50% faults in electrical motors are bearings-related [2]. Energy consumption, revolutions per minute, temperature, air gap eccentricity, vibration, and bearing health are some of the useful data claimed by various sensor manufacturers in relation to electric motors. This type of information can be useful in troubleshooting failed motors, inspecting the condition of operational motors, and determining when a motor requires a closer look or maintenance and to reduce the electric motor downtime.

The usage of IoT to collect data and analyze the sensed data helps to obtain knowledge from the raw data. Such information can be used to achieve predictive maintenance of electric motors. Some of the applications are turbines, paper mills, refrigeration, to name but a few.

1.1.3 Wireless sensor network (WSN)

Wireless sensor network (WSN) is one of the enabling technologies of Internet of Things. WSN is defined as a network of distributed sensor nodes which performs the task of sense, compute, and communicate. The WSN is an example of infrastructureless, short-range, personal area network. The communication standard for WSN is IEEE802.15.4. The WSN supports different types of topology such as star and mesh. The network comprises source nodes to monitor the environmental factors and forward the sensed data toward the sink node via multihop communication.

An approach for routing IPV6 packets over zigbee-based WSN is called the 6lowpan (IPV6 over low-power wireless personal area networks) [4]. The IP packets are compressed using the adaptation layer to make it suitable for the personal area network. With the help of border gateway, which acts as an interface between internet and the nodes in the sensor networks, the packets have been adapted suitable for Internet and wireless sensor networks.

1.1.4 Internet of Things (IoT)

Research trends in pervasive computing technologies have the principle of integrating the paradigm of many recent technological solutions for developing anytime, anywhere accessible devices and systems. This requires a major backbone technology as IoT. The master's and doctoral learning community, while exploring the design and applications of IoT, needs to establish new frameworks and integrate various ideas.

Internet of Things (IoT) leads in the world news today due to its wider potential of applications such as smart cities, smart homes, wearables, automobile industries, etc. Research in the field of IoT cannot be confined to a specific area [5]. It is enabled by handshaking of several domains of research such as sensors, networking, cloud computing, edge computing, big data, machine learning, and deep learning. As IoT is a technology outcome of multidisciplinary research, today's researchers are in need to develop Proof-of-Concept (POC) solutions on various aspects of IoT. There exists a significant tool for the design and development of IoT networks and solutions. For example, Contiki OS is a platform that has well-structured functions and modules supporting various design aspects of IoT networks. Usage of the features of communication stack of such tools provides extended IoT applications. The network layer of IoT can be enhanced with scheme of use of network coding at packet level to improve the throughput performance of IoT networks. Similarly, the security in the network layer of IoT can be enhanced by developing privacy homomorphism. Also such design can be enhanced with improvised Routing Protocol for Low-Power and Lossy Network (RPL) in the network layer for higher performances. Various applications can also be developed by generating new suitable functions of various algorithms as embedded solutions. **Figure 1** shows the architecture of Internet of Things where the hardware components are interconnected by the means of Internet. The following are the various benefits of IoT:

a. Communication between devices: IoT achieves the communication between devices, also famously known as Machine-to-Machine (M2M) communication. The devices are connected to a network, and hence, the control and transparency are available with greater efficiencies and quality.

b. Automation and control in working: As the devices are connected to a network, the devices can be controlled in a centralized manner with a widely used wireless technology called Wireless Fidelity (Wi-Fi).

c. Improved quality in monitoring of devices: IoT allows automating the tasks with less human intervention. The continuous monitoring leads to improved quality in decision-making, transparency, and quick decision-making during emergency situations.

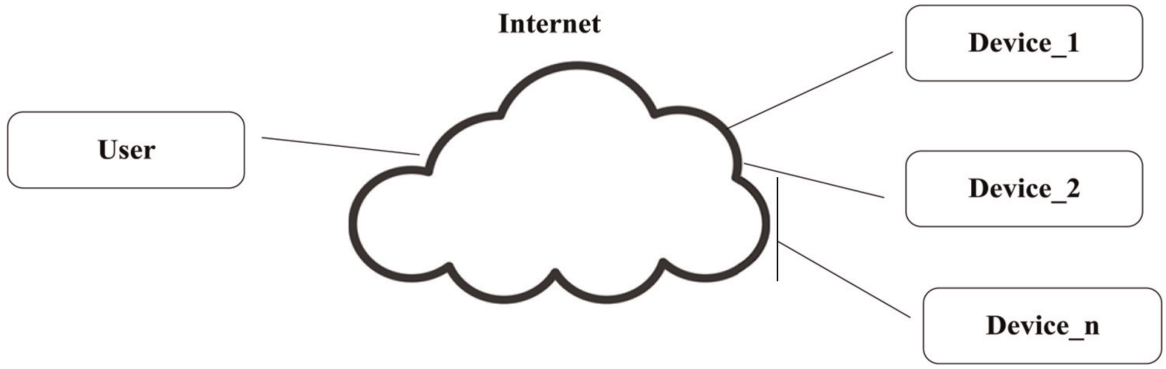

Figure 1.
Architecture of IoT.

2. Fault recovery analysis of induction motors

The induction motor finds application in most of the industries. Bearing fault in induction motor leads to more severity if not rectified in the initial stage [6]. The occurrence of bearing fault causes increased vibration and temperature of the motor. When the vibration goes beyond a certain level, it affects the air gap between stator and rotor and induces faulty frequency into the stator current. Many researchers have analyzed the faults in induction motors and proposed different strategies for monitoring and diagnosis. The following are few such existing solutions.

2.1 Online motor condition monitoring system for abnormality detection

An online motor condition monitoring system based on Cortex-M4 microcontroller with a graphic user interface is used for abnormality detection [7]. The system monitors the electrical and vibration signal for fault detection. The parameters monitored are voltage, current, and vibration. The captured signals are given to an infinite impulse file, and then fast Fourier transform is applied for spectral analysis to identify any abnormality in the captured signals pattern.

2.2 An analytical approach of parametric monitoring of induction motor using GSM

An embedded system based on ATMEGA-16 with Global System for Mobile Communication (GSM) has been used to protect the induction motor against overvoltage, overcurrent, and over-temperature [8]. The components such as timer, contactor, voltage, and current relays are used. The parameters used for finding the fault in the system are voltage, current, speed, and temperature. The parameters associated with the induction motor are collected for every periodical interval, the data are transmitted over GSM, and the messages are displayed in a Liquid Crystal Display (LCD) on the receiver end. Also the values are displayed on the mobile phone associated with the devices.

2.3 Acoustic based on fault diagnosis in induction motor

The work in [9] discusses an acoustic-based condition monitoring and fault diagnosis-based review to detect four different types of faults such as bearings, rotors, stators, and compounds. Various datasets are being analyzed using various machine learning algorithms. The type of fault determination in the induction motor is affected by environmental noise.

2.4 IoT-based vibration monitoring

The accelerometer sensor, which was mounted on the engine's rod axis and linked to a wireless RF device, was carried in different environments for different rotational speeds [10]. Furthermore, various types of vibration signals with varying amplitudes and frequencies are injected directly on the engine's axis to test and prove device reliability. Allan's variance technique allows for the successful detection, definition, and localization of vibration signatures.

2.5 Health monitoring using IoT and machine learning

A real-time machine health monitoring system that can analyze the supply balancing condition on a three-phase system by combining machine learning and IoT technology is proposed in [11]. This system is built with current transformer to capture and send electrical data from the load to the server. The server processes data by artificial neural network to train the data and for load classification.

2.6 Case study on fault diagnosis in induction motor

Incorporation of machine learning algorithms to aid or to take decision on its own on different types of faults in induction motor. In [12], different artificial intelligence algorithms and its suitability for fault identification have been discussed.

- Neural network, after the training phase, is used to classify the incoming data. The value that lies outside the range is named as a potential motor fault. To avoid false fault diagnosis, the alarm is raised when fault value ranges are observed persistently. It is suitable to diagnose bearing and unbalanced rotor faults of induction motors.

- Fuzzy-logic-based systems have been used to classify broken-bar-related. A set of nine rules are used to determine the two sideband components. The broken bars are identified based on the sideband components.

- A spectral kurtosis and envelope spectrum to identify different types of faults in rolling element bearings [13]. The dataset [14] contains an acceleration signal “gs,” sampling rate “sr,” shaft speed “rate,” load weight “load,” and four critical frequencies representing different fault locations such as ballpass frequency outer race, ballpass frequency inner race, fundamental train frequency, and ball spin frequency.

2.7 Inference

Fault detection and diagnosis are an aiding tool for the accurate determination of different types of fault in induction motor. The efficiency of the fault diagnosis system depends on the system or algorithm accuracy. Fault prediction is capable of predicting early possible development of fault in the induction motor. It leads to reduced maintenance cost and less shut down time of the equipment. Fault prediction system in association with Internet of Things could be an effective method to continuously track the status of the equipment and allocate periodical prior schedule of the equipment from a remote location.

3. Fault prediction model

The prediction model is designed with Naïve Bayes algorithm. Naïve Bayes classifier predicts that the presence (or absence) of a particular feature of a class is unrelated to the presence (or absence) of any other feature [15]. This classifier is very simple, efficient, and is having a good performance. Sometimes it often outperforms more sophisticated classifiers even when the assumption of independent predictors is far. This advantage is especially pronounced when the number of predictors is large.

3.1 Overall system

The system is designed with the objective of identifying the fault condition in the induction motor and informing the status of the induction motor to the lab in-charge in a remote fashion. The advantage of this approach is that the induction motors located at different premises are monitored, and periodical maintenance is allocated in a centralized manner as shown in **Figure 2**. The induction motor equipped with accelerometer to monitor the vibration data and the data are sent to the gateway node in a wireless fashion. The gateway node forwards the data to the control room for every periodical interval. The decision-making software runs the Naïve Bayes algorithm on the received data. The algorithm predicts the possible occurrence of fault. The alarm will be given to the lab in charge for further maintenance if any.

Figure 3 shows the proposed prediction technique. The induction motor is connected with accelerometer. The accelerometer data are forwarded to the gateway and to the control room server. The server runs the Naïve Bayes algorithm for the purpose of occurrences of fault. The acceleration data are processed by the application created by python programming language. The Naïve Bayes algorithm implemented in python language has been trained with the accelerometer data. After testing with different samples of data with both normally working machine and with faulty induction motor, the predictor module predicts the status of the machine from incoming real data after during the real-time running of the motor. The outcome of the predictor is divided into two classes, namely normal and faulty class. If faulty class is predicted, then an alert is given to the lab in-charges. Hence, preventive maintenance is carried out to avoid long time stoppage of the motor, thus improving the production time of the induction motors.

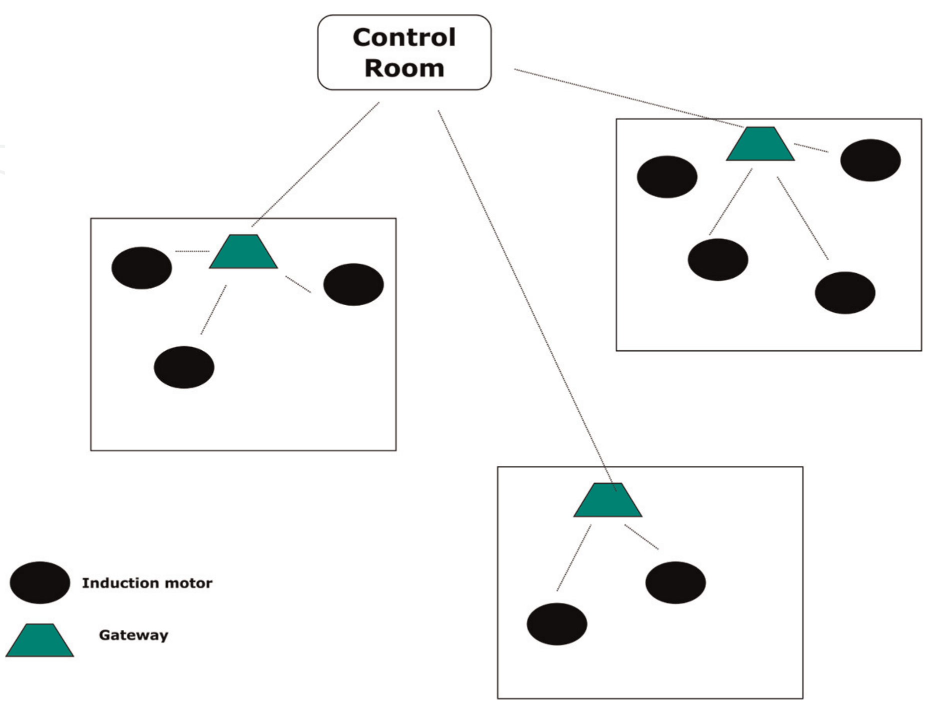

Figure 2.
Overall system setup.

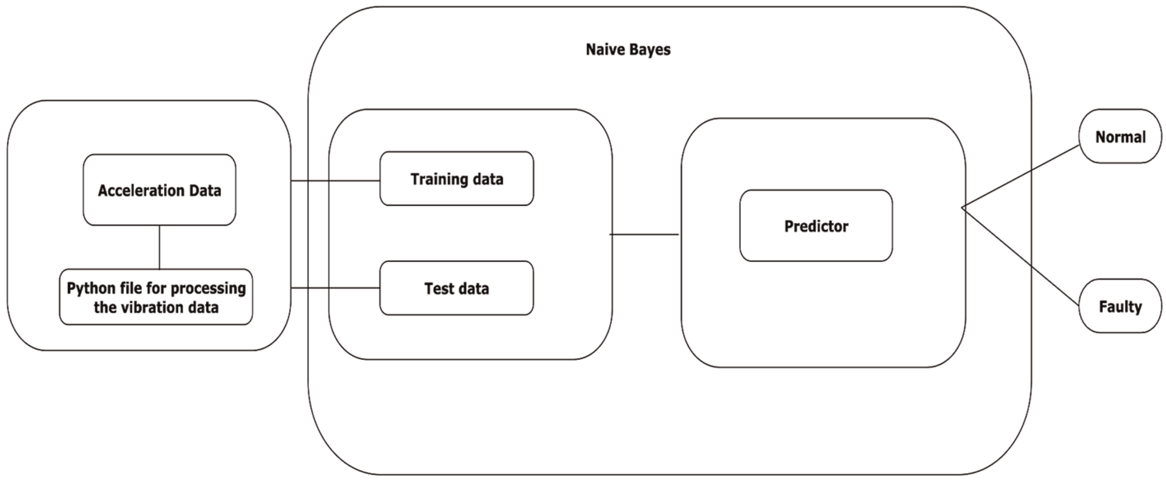

Figure 3.
Proposed decision-making technique.

3.2 Modules of prediction model

Figure 4 shows the module of the proposed technique. The detailed explanation of the modules is given in this section.

Figure 5 shows the experimental setup for measuring the vibration of the induction motor. An embedded base board with Wi-Fi support known as Intel Edison is connected with the accelerometer using the Arduino's Uno board, which acts as data acquisition unit. The vibration data from the induction motor are collected for every periodical interval. The sensed data are forwarded to the lab in-charge, to know the status of the machine. Also, the data are processed with Naïve Bayes algorithm for the determination of bearing fault.

Based on the outcome of the decision-making module, which runs the Naïve Bayes algorithm, the lab in-charge decides whether to allow the motor to run or to stop or halt the motor for a specific period of time. The hardware and software used for the implementation are tabulated in **Table 1**.

The embedded device we have used in the work is Sparkfun Intel Edison board [16]. It is a lightweight board designed to support Internet of Things, and the base board consists of 70 pins. It has inbuilt Wi-Fi and bluetooth, it supports Yocto Linux operating system. The operating voltage of the board is 3.3–4.5 V. The application development could be done by Python, Jjava, Node.js, C, C++. The application development on the Intel Edison could be done by performing the connection via putty software.

The Intel Edison board is connected with Arduino Uno using Inter Integrated Circuit (I2C) protocol as shown in **Figure 6**. The Intel Edison board is the master, and the Arduino Uno is the slave. The reason for adapting Arduino Uno is its easy interfacing support with sensors. I2C is a synchronous serial protocol, the clock signal is controlled by the master as shown in **Figure 6**. The serial data (SDA) line is used for the master and slave to send data. The serial clock (SCL) line carries the serial clock. In I2C, the messages are broken into frames. Each message has an address frame that contains the binary address of the slave, and one or more data frames that contain the data being transmitted. The message also includes start and stop conditions, read/write bits, and ACK/NACK bits between each data frame.

Figure 7 shows the interfacing of ADXL335 with Arduino. ADXL335 has physical vibrations as input and the three-axes analog output is taken via the X, Y, and Z pins from the accelerometer. The Arduino UNO has A1, A2, and A3 as analog input/output

DOI: http://dx.doi.org/10.5772/intechopen.103727

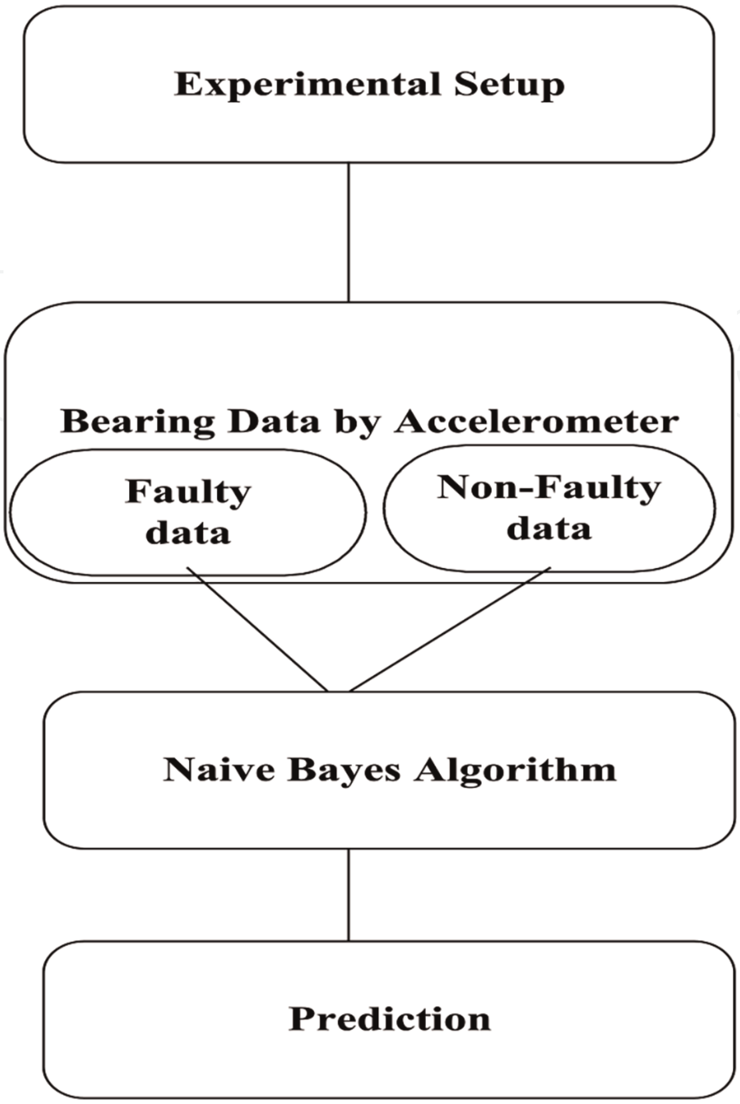

Figure 4.
Modules of proposed technique.

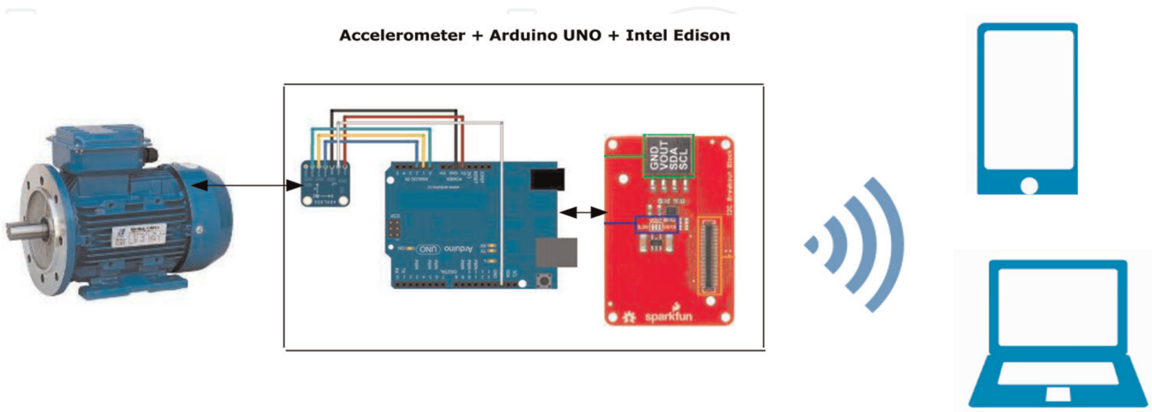

Figure 5.
Experimental setup.

pins. Accelerometer output is given as input to the Arduino by connecting X, Y, and Z to A1, A2, and A3, respectively. The Aref pin of the Arduino is shorted with the 3.3 V pin of the Arduino itself. Accelerometer receives supply from the Arduino by connecting the Vcc and GND pins of the both with each other.

Hardware	Software
Single-phase induction motor	Python (Application and Device Driver)
ADXL 335 3-axis Accelerometer	Arduino Code (Dialect of C/C++)
Arduino UNO	HTML
Sparkfun Intel Edison	Naïve Bayes algorithm

Table 1.
Components for vibration monitoring.

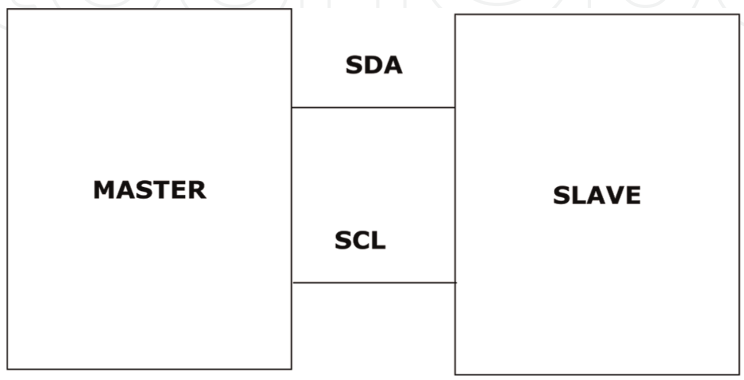

Figure 6.
I2C communication between master and slave.

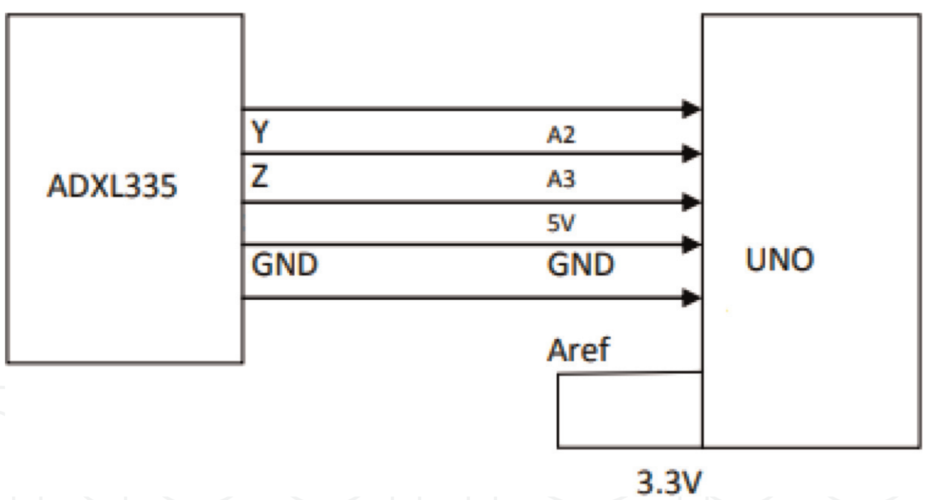

Figure 7.
Interfacing accelerometer sensor with Arduino Uno.

Figure 8 shows the interfacing of Arduino Uno with Intel Edison. The accelerometer values are taken to the Intel Edison board via I2C protocol. A4 and A5 pins of the Uno are the analog input/output pins. I2C board block of the Intel Edison block has SCL and SDA pins for serial communication. A4 and A5 of the Arduino are connected to SDA and SCL, respectively. The GND-GND connection of both the board needs to be ensured.

- Prediction analysis and solution: Naïve Bayes classifier

Predictive maintenance is a method to predict the occurrence of fault in the system in advance. The advantage of such approach is the time and cost involved in

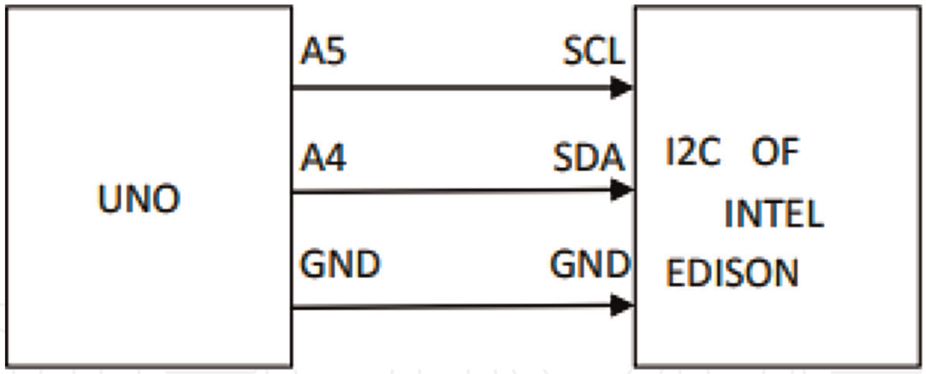

Figure 8.
Interfacing Arduino Uno with Intel Edison.

overcoming the fault is less compared with the traditional approaches such as run to failure and preventive maintenance. There are different prediction algorithms used to determine the fault or anomaly in advance. In [17], comparison of supervised machine learning algorithms for IoT data has been performed. Algorithms such as Naïve Bayes, decision tree, random forest, k-nearest neighbor, and logistic regression are being used for different datasets. The summary is as follows:

- The Naïve Bayes is suitable for moderate size of dataset.
- As the number of features increases, the time taken for the result is increased in all algorithms.
- For better performance, the number of classes needs to be kept minimal in Naïve Bayes, whereas in decision tree, random forest, and k-nearest neighbor, the number of classes do not have impact on the performance.
- Naïve Bayes performs well even with missing values in the dataset.

The advantages of Naïve Bayes algorithm are as follows:

i. Simple and fast

ii. Requires less training data

iii. Suitable for continuous and discrete data

iv. Suitable to make probabilistic predictions

The Naïve Bayes classifier is based on Bayes theorem and is used to find the probability of each instance belongs to a specific class. It operates on dataset X with attribute values $\{x_1, x_2, \ldots, x_n\}$ and determines the target function Y(X) from a predefined set $S = \{s_1, s_2, .., s_n\}$

$$P(S|X) = \frac{P(X|S)\,P(S)}{p(X)} \tag{2}$$

$P(\mathrm{S})$ is the prior probability about the class S. $P(\mathrm{X}|S)$ is the posterior probability of X when the class S has given. The probability of obtaining result X without knowing S has occurred is $p(\mathrm{X})$, and $P(\mathrm{S}|X)$ is the posteriori probability of X.

4. Simulation results

To evaluate the performance of the prediction system, we use python to develop the application for Internet of things, device driver code, and also the Naïve Bayes algorithm. To conduct the simulation, we used the dataset with and without fault. The dataset has been prepared by running the induction motor with and without bearing faults for different time intervals. The acceleration data were collected for an interval of 20 s. Some of the collected data are used for training to determine the threshold using Naïve Bayes model.

To classify the class from the new data, we have chosen the faulty value (1 or 0)

$$classifier(\mathrm{x}_1, \mathrm{x}_2, \dots, \mathrm{x}_\mathrm{n}) = \underset{s \in \{0,1\}}{\operatorname{argmax}} P(\mathrm{x}_1, \mathrm{x}_2, \dots, \mathrm{x}_\mathrm{n}|S) \tag{3}$$

The relative frequency counting according to multinomial Naïve Bayes is given as

$$P(X|S) = \frac{N_{xi,si} + a}{N_{si} + ad} \tag{4}$$

where $N_{xi,si}$ is the number of times the features x_i appears in class s_i; a is the parameter for additive smoothing; d is the dimensionality of the feature vector.

4.1 Data collection setup

The vibration data are collected using accelerometers, the vibration at different positions of the motor housing is studied. The position near the shaft is chosen for measuring the vibration. The vibration values are collected for different scenarios such as no load and moderate load. The accelerometer data are collected and forwarded to the server using the Intel Edison embedded board. Ten datasets are used in this study, which includes normal and faulty scenario. The faults may be any of the types such as inner race, ball defect, train defect, and outer race. All the 10 datasets are separated to form the new dataset in a fixed length of 800 after performing data cleanup. The label of the samples $\mathrm{Y} = \{0, 1\}$ and the definition of the element are shown in **Table 2**.

4.2 Results

Table 3 shows the overall accuracy of the prediction algorithm to predict the possibility of fault in induction motor. Under no load scenario, the fault prediction is

Label	Status
0	Normal
1	Possibility of fault

Table 2.
Labels for classes.

	No load	Load
Accuracy	100%	93.2%

Table 3.
Accuracy.

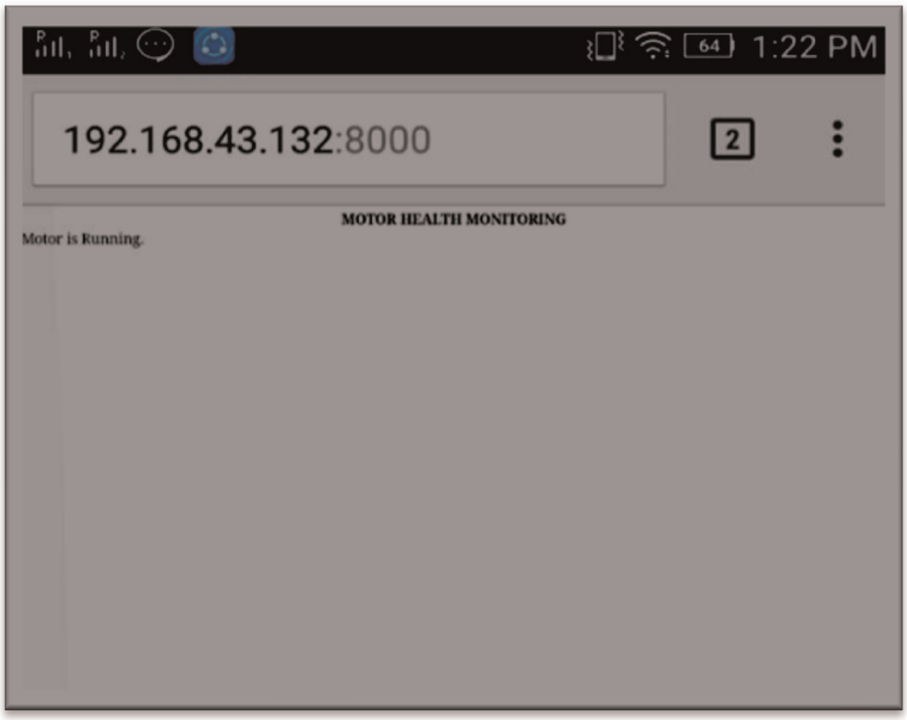

Figure 9.
Status of motor observed using an URL.

accurate, whereas during running condition, the accuracy is reduced, it could have been avoided by increasing the training samples and to have dataset collection for different variants of load, as we focus on the presence of fault or not, we are done with a moderate level of data for training and prediction.

Figure 9 shows the status of the motor in a web page using a dedicated address. The web page is updated with the status of motor running; if any fault is expected, an update is given to the server and the web page of the lab in-charge is automatically updated.

5. Conclusions

This chapter presented the design and validation of an IoT system for monitoring the health of induction motors in a laboratory setting. The system incorporates the use of a sensor in the IoT board that wirelessly transmits the sensed data, and the status of the induction motor after evaluation with the machine learning algorithm known as Naïve Bayes is made available on the web page. The accuracy of the developed system was evaluated under no load and load conditions.

Future work in this domain is to extend the work with more nodes and to consider many induction motor parameters to predict the fault in the induction motors by using a bag of learning model.

Conflict of interest

The authors declare that they have no conflict of interest.

Author details

A. Pravin Renold[1]* and B. Venkatalakshmi[2]*

1 Mepco Schlenk Engineering College, Sivakasi, India

2 L&T EduTech, Chennai, India

*Address all correspondence to: pravinrenold@mepcoeng.ac.in
and venkatalakshmi.71@gmail.com

References

[1] Vas P. Parameter Estimation, Condition Monitoring, and Diagnosis of Electrical Machines. New York: Oxford University Press; 1993

[2] Kliman GB, Stein J. Induction motor fault detection via passive current monitoring. In: International Conference in Electrical Machines. August 13, 1990; Cambridge, MA. 1990. pp. 13-17

[3] Chen S, Lipo TA. Bearing currents and shaft voltages of an induction motor under hard-and soft-switching inverter excitation. IEEE Transactions on Industry Applications. 1998;**34**(5): 1042-1048

[4] Rana B, Singh Y, Singh PK. A systematic survey on internet of things: Energy efficiency and interoperability perspective. Transactions on Emerging Telecommunications Technologies. 2021;**32**(8):e4166

[5] Armentano R, Bhadoria RS, Chatterjee P, Deka GC, editors. The Internet of Things: Foundation for Smart Cities, EHealth, and Ubiquitous Computing. Boca Raton, Florida: CRC Press; 2017

[6] Kompella KD, Rao MV, Rao RS. Bearing fault detection in a 3 phase induction motor using stator current frequency spectral subtraction with various wavelet decomposition techniques. Ain Shams Engineering Journal. 2018;**9**(4):2427-2439

[7] Chang HC, Jheng YM, Kuo CC, Huang LB. On-line motor condition monitoring system for abnormality detection. Computers & Electrical Engineering. 2016;**51**:255-269

[8] Lande S, Jaiswal P, Rajgure P. An analytical approach of parametric monitoring of induction motor using GSM. IOSR Journal of Electronics and Communication Engineering. 2012;**1**(3): 01-07

[9] AlShorman O, Alkahatni F, Masadeh M, Irfan M, Glowacz A, Althobiani F, et al. Sounds and acoustic emission-based early fault diagnosis of induction motor: A review study. Advances in Mechanical Engineering. 2021;**13**(2):1687814021996915

[10] Hayouni M, Bousselmi Z, Vuong TH, Choubani F, David J. Wireless IoT approach for testing in situ motor's axis vibration monitoring. In: 2021 International Wireless Communications and Mobile Computing (IWCMC); 2021 June; IEEE. 2021. pp. 92-97

[11] Wong TK, Mun HK, Phang SK, Lum KL, Tan WQ. Real-time machine health monitoring system using machine learning with IoT technology. In: MATEC Web of Conferences, Vol. 335. EDP Sciences. 2021. p. 02005

[12] Nandi S, Toliyat HA, Li X. Condition monitoring and fault diagnosis of electrical motors—A review. IEEE Transactions on Energy Conversion. 2005;**20**(4):719-729

[13] Matlab [Internet]. 2020. Available from: https://in.mathworks.com/help/predmaint/examples/Rolling-Element-Bearing-Fault-Diagnosis.html

[14] Bechhoefer E. Condition based maintenance fault database for testing diagnostics and prognostic algorithms. 2013. Available from: https://mfpt.org/fault-data-sets/ [Online accessed Oct-2021]

[15] Soni J, Ansari U, Sharma D, Soni S. Predictive data mining for medical

diagnosis: An overview of heart disease prediction. International Journal of Computer Applications. 2011;**17**(8): 43-48

[16] Sparkfun Intel Edison [Internet]. 2020. Available from: https://learn.sparkfun.com/tutorials/general-guide-to-sparkfun-blocks-for-intel-edison/all

[17] Khadse V, Mahalle PN, Biraris SV. An empirical comparison of supervised machine learning algorithms for internet of things data. In: 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA); August; IEEE. 2018. pp. 1-6