Journal Européen des Systèmes Automatisés
Vol. 58, No. 2, February, 2025, pp. 373-381
Journal homepage: http://iieta.org/journals/jesa

A Predictive Maintenance System Based on Industrial Internet of Things for Multimachine Multiclass Using Deep Neural Network

Ruaa W. Abdalah[1*], Osamah F. Abdulateef[1], Ali H. Hamad[2]

[1] Automated Manufacturing Engineering Department, AlKhwarizmi College of Engineering, University of Baghdad, Baghdad 10071, Iraq
[2] Information and Communication Engineering Department., AlKhwarizmi College of Engineering, University of Baghdad, Baghdad 10071, Iraq

Corresponding Author Email: ruaa.abd2204@kecbu.uobaghdad.edu.iq

https://doi.org/10.18280/jesa.580218

Received: 4 December 2024
Revised: 25 January 2025
Accepted: 9 February 2025
Available online: 28 February 2025

Keywords:
predictive maintenance (PdM), IIoT, sensor, DNN, multiclass, custom dataset

ABSTRACT

Among the many applications of Industry 4.0, predictive maintenance is one of the most frequently utilized examples. On the other hand, in order to improve failure categorization, the majority of contemporary machine-learning models require a substantial amount of data. In contrast to traditional maintenance, IIoT systems that perform real-time monitoring can be of tremendous service to the company. These systems can notify the necessary members of the factory's maintenance team in advance of a serious breakdown, which offers a significant advantage. It is of the utmost importance to detect any malfunctions in equipment while they are in operation before they become critical. The purpose of this work is to collect a substantial quantity of data from three AC motors, each of which is equipped with four different kinds of sensors. These sensors include a vibration sensor, a current sensor, a contactless temperature sensor, and an ambient temperature sensor. A variety of motor faults, including normal, vibration, stop, heavy load, and overcurrent, have been purposefully applied to the system in order to build the custom dataset. These motor's faults have been categorized and labeled in accordance with their respective classification responsibilities. A deep neural network (DNN) model consisting of seven layers was utilized. A cloud server is used to train the model, and all of the data from the three AC motors are sent to the cloud server after they have been collected. The result demonstrates that it has good accuracy and loss in both the training and testing phases, with a loss of 0.0014 and an accuracy of 100% while the model has been tested for over and under fitting problems.

1. INTRODUCTION

In the present-day global economy, which is more globalized and devolving further to dematerialized markets, industries are driven to improve the productivity of their value chains, increasing their competitiveness and satisfying customers. Connectivity leads to data production or generation, the availability of new devices, the reduction in inventories, customization, and regulated manufacturing have given birth to Industry 4.0, which seems inevitable. This suggests that automation techniques must be used to incorporate all new technology in order to significantly boost production [1]. The Internet of Things (IoT) is a rapidly expanding network of devices, sensors, and ubiquitous communications that significantly impacts a wide range of fields. Smart industrial operations can be made possible by increasing the speed and volume of data generated by Industrial Internet of Things (IIoT) devices combined with advanced analytics capabilities, especially in the context of Industry 4.0. IIoT represents a new vision of the IoT by automating smart objects to perceive, collect, analyze, and share events in real-time. The key benefits of IIoT are enhanced operational effectiveness, enhanced production rate, and management of industrial assets and processes involving product differentiation, smart monitoring apps, as well as machine health checks and preservation [2].

There are four categories of machine maintenance: Predictive maintenance (PdM) detects potential issues prior to their manifestation to enhance equipment durability and avert unforeseen machine downtime. Proactive maintenance (PRM) targets the underlying causes of failures to avert their occurrence, whereas reactive maintenance (RM) concentrates on repairing machines just after they malfunction. This indicates that downtime could be considerable due to the uncertain duration of failures. Therefore, preventative maintenance (PM) is performed at regular intervals regardless of the machine's actual state.

Unlike reactive, which is less expensive but causes unanticipated downtime and higher costs should an equipment break down. While preventative measures may lead to unnecessary maintenance, predictive measures reduce unscheduled downtime and allow planning. PdM involves condition monitoring, so it only happens when necessary. This should cut the demand for spare parts and help to reduce costs.

It also helps to avoid too much upkeep, which often causes issues. While PM may involve reevaluating systems to prevent breakdown, PdM is mostly focused on arranging repairs based on the actual state of the unit. Further advantages come from applying modern technologies such as IoT sensors and data analysis machine learning. Forecasts thus become more accurate. Early problem fixes can help to extend equipment life.

PdM requires data storage and analysis in real time, taking into account the different aspects and effects of the collected signals. PdM uses a number of mathematical techniques, including machine learning, reliability analysis, time series analysis, statistical process, and failure mode and effects analysis [3]. The foundation of PdM is based on the use of data analytics tools to interpret operational data transmitted by sensors [4]. Condition monitoring systems typically use data generated by sensors directly connected to industrial machinery to identify anomalies and predict failures, thereby improving reliability and enabling cost-effective maintenance. Condition monitoring uses a wide range of sensor components, such as temperature and vibration sensors, to monitor various system attributes across multiple domains, simulate system behavior, and identify anomalies in the data that indicate damage or deterioration. This is particularly important because failures can have a significant and cascading impact on the overall process or industrial production cycle, potentially resulting in expensive maintenance costs and lost revenue [5]. Machine learning and deep learning, two recent developments in the AI area, have been shown to be successful methods for creating PdM models because of their capacity to carry out failure prediction tasks like calculating a machine's remaining usable life [6, 7]. Because of the vast system architectures of industrial equipment made of many elements, an advanced maintenance model is required. Even when the physical and digital systems become integrated within the manufacturing process, it becomes possible to gather large amounts of data. This is because of deep learning, which employs neural networks with several layers of processing units [8]. Deep learning is used in many disciplines today, including PdM. Deep learning models perform better than statistical and traditional machine learning models when sufficient historical data is available. Artificial neural networks (ANN), a machine learning technique inspired by brain activity, go beyond shallow networks with 1 and 2 hidden layers, which is referred to as deep learning [9]. Sensors are installed on machines to collect data about their condition on a continuous basis. The data from the machines is used to find trends that lead to early detection of faults by connecting the machines to the sensors through the IoT [10].

In this work, a PdM system based on a DNN has been proposed and implemented practically on three AC motor benchmarks, which are occupied with four types of sensors. A large amount of data has been collected offline to train the DNN model on the cloud server, while in the online phase, the data is collected and uploaded to the model to predict the failure type.

The rest of this paper is as follows: section two shows the state of the art of the proposed system design. Section three introduces the premieres required to design the proposed system. The proposed system design is described in section four, while the results are discussed in section five. Finally, the conclusion is presented in section six.

2. RELATED WORK

A great deal of research has been done on artificial intelligence (AI) for PdM systems using various modeling techniques. Drakaki et al. [11] developed a deep learning and mathematical programming-based integrated PdM and production planning framework to reduce the total cost of manufacturing, setup, holding, maintenance, and backorder. The paper's usefulness is limited since it does not adequately address the integration issues of machine learning and deep learning techniques in current industrial systems. Gatta et al. [12] developed a deep learning model for diagnosing and prognosticating issues in offshore oil wells using the public 3 W dataset. It employs a one-dimensional Convolutional Neural Network (1D CNN) as an encoder for feature extraction, comparing it with traditional statistical methods. Various machine learning algorithms, including Random Forest Classifiers, are utilized to evaluate the extracted feature.

Es-Sakali et al. [13] introduced a prototype that would enable staff members to keep an eye on the machine's condition and anticipate any issues before the planned maintenance date, preventing unplanned malfunctions and a halt in production. Mohammed et al. [14] designed an electrical motor based on IoT and machine learning with the goal of increasing reliability and lowering maintenance expenses. PRM procedures are made possible by utilizing real-time data. However, obstacles, including implementation difficulty, cybersecurity concerns, and reliance on data quality, might prevent wider use. The significant initial cost and possible problems with model interpretability are also significant disadvantages. Zainol and Burhani [15] introduced a feature selection and deep learning algorithms to forecasting motor bearing faults accurately. Three distinct classifiers were used: Radial Basis Network (RBN), Feedforward Neural Network (FFNN), and Convolutional Neural Network (CNN). In order to categorize motor performance and detect issues like inner and outer race problems, several classifiers were employed. Performance Results: In CNN testing, the suggested technique obtained maximum accuracy rates of 95.4% and 97.7%, indicating the model's efficacy in classification. Karwa et al. [16] developed a machine learning algorithm to analyze the most influential factor of maintenance systems in a AC motor or induction motor. The aim of this work is to create an efficient machine learning algorithm for high voltage induction motor failure. Proposed model is an ANN and three sensors: power factor vibration, and temperature. The model was correlated with the sensors to improve the selected factor for maintenance.

Al-Naggar et al. [17] proposed IoT technology for PdM and track the health of four CNC machines spread across multiple sites. Accelerometers collect vibration signals and transmit real-time data directly to a database. The results show that analysis of the acceleration signal in the time and frequency domains can successfully identify the status of each machine, enabling simultaneous monitoring across multiple sites. A potential limitation of the article is that it may not adequately address the privacy and data security issues associated with IoT deployments in industrial environments. Singh and Singh [18] presented identified a motor performance and forecast motor bearing problems using deep learning classifiers, such as CNN, FFNN, and RBN. The study highlights the value of feature selection strategies like chi-square and correlation to improve prediction accuracy, with tests showing up to 97.7% accuracy. Karthikeyan et al. [19] introduced a deep learning-based PdM system for industrial rotating machinery that estimates remaining usable life (RUL) without the need for labeled data by using a regression model and an LSTM autoencoder to identify abnormalities. Regression analysis and the LSTM autoencoder are used in this approach to improve flexibility and dependability in practical applications. Furthermore, operational data is gathered via a wireless condition monitoring system and sent for analysis. This strategy seeks to increase machine dependability in a variety of industrial contexts and decrease unplanned downtime.

3. PRELIMINARIES

In order to enable preventive maintenance, PdM makes use of IIoT and data analysis to track the health of equipment and anticipate problems. This work creates a PdM system based on deep learning and the IIoT.

3.1 Sensors

Four sensors have been used in the proposed PdM system. The ADXL345 is a perfect gadget for measuring motor vibrations as it is a 3-axis shake sensor that is 13-bit. It has nice properties such as a small form factor with low power consumption. The current in the motor is monitored using the ACS712 current sensor module. It is a completely integrated magnetically hysteresis-free current sensor. It uses the Hall phenomena effect to provide current sense for both DC and AC currents, and it runs on a single 5 V power source. It also makes use of an ADS1115 analog-to-digital converter and a low-resistance current conductor. Specific important features must be had by the temperature sensor used for this investigation was the MLX90614 infrared thermometer. It was selected because of its wide range temperature, excellent precision, and contactless temperature monitoring ability. It also fits the needs of this research nicely because it is affordable and small in size. Temperature readings from the outside were also taken using an SHT21 digital humidity and temperature sensor [20].

3.2 Inter-integrated circuit (I2C protocol)

Raspberry Pi is suitable platform for IoT applications due to its portability, parallelism, affordability, and low power consumption. IoT applications use raspberry for its low cost and low power computation that can perform many of the same tasks as a typical desktop computer. Raspberry is a quad-core processor with parallel processing can be used to accelerate up operations and computations [19]. To connect sensors to Raspberry Pi, a 12IC Bus is used, which a two-wire interface is using a key design to facilitate communication between multiple integrated circuits (ICs). It connects sensors to Raspberry via only two wires (SCL and SDA). This makes it easy for Raspberry to control the data sent and received from the sensors [21].

3.3 Dataset implementation

In this work, the sensory data was collected using sensors, which are the current sensor, non-contact temperature sensor, temperature and humidity sensor, and three-axis accelerometer sensor. When the vibration sensor generates three sets of data in three axes: X, Y, and Z. These results are in the same column of the dataset, and there'd be a fault condition every 1 second of sampling time. The data from the sensor is collected both online and offline, where offline, a dense amount of data is collected with fine classes of fault, namely: normal, overcurrent, stop rotation, misalignment, and heavy load. These classes were intently applied in the motor to obtain the required dataset according to the required fault. An unbalanced mass has been added along with the motor shaft to simulate the vibration in the three axes of the motor. A different size of the capacitor is used to simulate the increase and decrease of the motor's current. A mechanical braking system was used to simulate the stop and heavy load conditions.

Table 1 displays the number and amount of data sets for each categorization. Table 2 shows examples of the data that were gathered. The deep learning model uses this dataset to create a prediction model which works on a cloud server for real time monitoring that anticipates any failures.

The collected dataset is usually divided into four groups. The model is trained using the training dataset, which makes up about 60% of the total data gathered. The validation dataset is 10%, which is used to assess the model's performance and adjust its hyper parameters, and the test data (unseen data) makes up 30% of the data.

Table 1. Classes counts and sizes

Failure Class	File Size (KB)	Count Raws
Normal	813	20059
Over-current failure	202	4591
Misalignment/vibration failure	301	6562
Stop rotating failure	651	16566
Havey load failure	140	2911

Table 2. Samples for the collected dataset

Accel_x	Accel_y	Accel_z	Amb_Temperature	Object_Temperature	AC Current	Label
1.21875	-1.78125	8.3125	17.811	31.13	0.331	Normal
1.78125	-1.15625	7.375	17.822	31.07	0.331	Normal
0.53125	0.125	7.5	17.822	31.01	0.331	Normal
-4.03125	5.28125	-16	35.175	36.17	0.426	Over current
-1.343	0.34375	-16	35.164	36.17	0.416	Over current
1.25	-0.875	7.0312	33.051	32.17	0.529	Stop
1.5625	-0.9375	7.4375	33.051	32.09	0.558	Stop
0.0404	-0.0269	-0.248	0.008	34.33	28.483	Misalignment
0.0429	-0.0465	-0.262	0.00764	34.21	28.483	Misalignment
-16	-16	-11.375	23.742	54.89	0.672	Heavy load
-16	-16	3.0625	23.731	54.79	0.689	Heavy load

3.4 Deep learning neural networks

The hyperparameters for the proposed DNN model are depicted in Table 3. The model consists of five layers. The input layer is equal to the input feature while the output layer is equal to the number of classes which are five too [22].

Table 3. Hyperparameter of proposed lightweight model

Parameter Name	Value
Neurons per layer	50
Batch_size	64
Epochs	25
Loss_function	Categorical cross-entropy
Activation_function	ReLU and SoftMax
Optimizer	Adam
Number of classes (output layer)	5

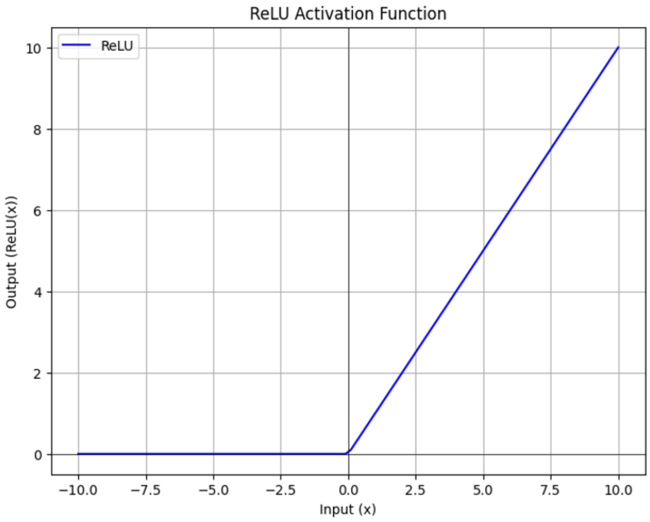

Figure 1. Rectified linear unit [23]

In a DNN, ReLU is typically employed as an activation function for the hidden layers. In order to do this, we utilize the neural network's penultimate layer activation to backpropagate the ReLU classification layer's weight parameters (Figure 1) [23]. ReLU formula is:

$$f(x) = \max(0, x) \tag{1}$$

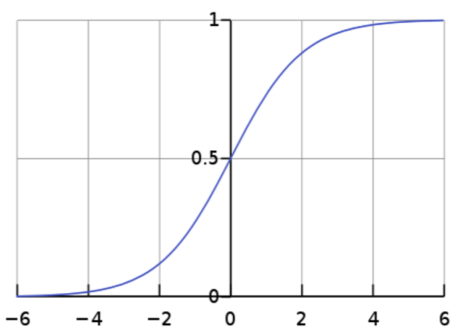

Figure 2. SoftMax activation function generates [23]

This function produces 0 when x is less than 0 and a linear function when x is greater than 0 (see Figure 2 for a visual illustration). Deep learning model for multiclass classification problems usually employ the SoftMax function as activation function (at the last layer). The SoftMax formula is as follows:

$$S(x_i) = \frac{e^{x_i}}{\sum_j^n = 1^{e^{x_j}}} \tag{2}$$

The SoftMax activation function converts a vector of K real values into a vector of real values that add up to 1. A probability score may be derived from the function's output, which is always in the range between 0 and 1. Due to its compatibility with the output format of SoftMax activation and one-hot encoding, categorical cross entropy (CCE) is frequently utilized as the loss function in multiclass classification applications. Each component fits together as follows:

1. Activation of SoftMax: In multiclass classification, the output layer of the model usually has one neuron for each class. The output logits (raw scores) are converted into probabilities using SoftMax, where each value denotes the likelihood of a class, and the probabilities for all classes add up to 1.

In categorical classification problems, when we want to select the single most likely class, SoftMax effectively emphasizes the class with the highest probability.

2. Encoding One-Hot: Multiclass jobs often use one-hot encoding for labels. Each class is encoded as a vector in one-hot encoding, with a `1` designating the proper class and a `0` designating all other classes. This facilitates the comparison between the real labels and the model's expected probability (SoftMax output) [24].

3. The concept of categorical cross entropy, or CCE.

By contrasting the one-hot encoded true labels with the SoftMax-predicted probability, categorical cross entropy determines the loss [25, 26].

For a single example, CCE is calculated mathematically as follows:

$$CCE = -\sum_{i=1}^{C} y_i . log\ (p_i) \tag{3}$$

where,

y_i is the one-hot encoded true label (1 for the correct class, 0 otherwise).

p_i is the predicted probability for each class.

The model is designed to generate higher-confidence predictions for the correct class because the loss reduces to the negative log of the predicted probability for the correct class since only the correct class contributes to the loss (because other (y_i) values are zero). In conclusion, categorical cross entropy is favored because it works well with one-hot encoding and SoftMax outputs. It also encourages the model to improve its predictions for multiclass classification problems by penalizing it when it assigns a low probability to the right class.

4. PROPOSED SYSTEM DESIGN

The proposed system of PdM consists of three AC motors (220V, 0.5A). Four sensors are connected to each motor, which are current, vibration, contact-less object temperature, and ambient temperature. These sensors monitor the condition of the motor on flying. A Raspberry Pi 4 is used to collect the data from the sensor using the I2C bus communication protocol. The function of the Raspberry Pi is to collect and upload the data to the cloud server. Two phases of operation were considered in this work: offline and online. Within the offline phase, as shown in Figure 3, data is collected from the sensor via the I2C protocol, and then this data is uploaded to the cloud server to generate a dataset from all the failure classes. In the cloud server, a deep neural learning DNN consists of seven layers of 50 neuron per layer; seven neuron at the input layer represent each column of the dataset, and five neuron at the output layer represent each class of the system. When the model is being trained, the online phase is started, where the sensor data is again collected from each motor and uploaded to the cloud server, where the DNN model predicts the failure type according to the uploaded pattern for each motor. Figure 4 shows the proposed schematic diagram for the I2C protocol to connect the sensors to the Raspberry Pi 4, SHT21, MLX90614, and ADXL345, which have a built-in I2C protocol, while ACS712 must be connected to an analog-to-digital converter ADC0X48, which converts the current value (AC) to digital form. The ADC has a built-in I2C. The I2c protocol consists of two lines: serial data (SDA) and serial clock (SCL).

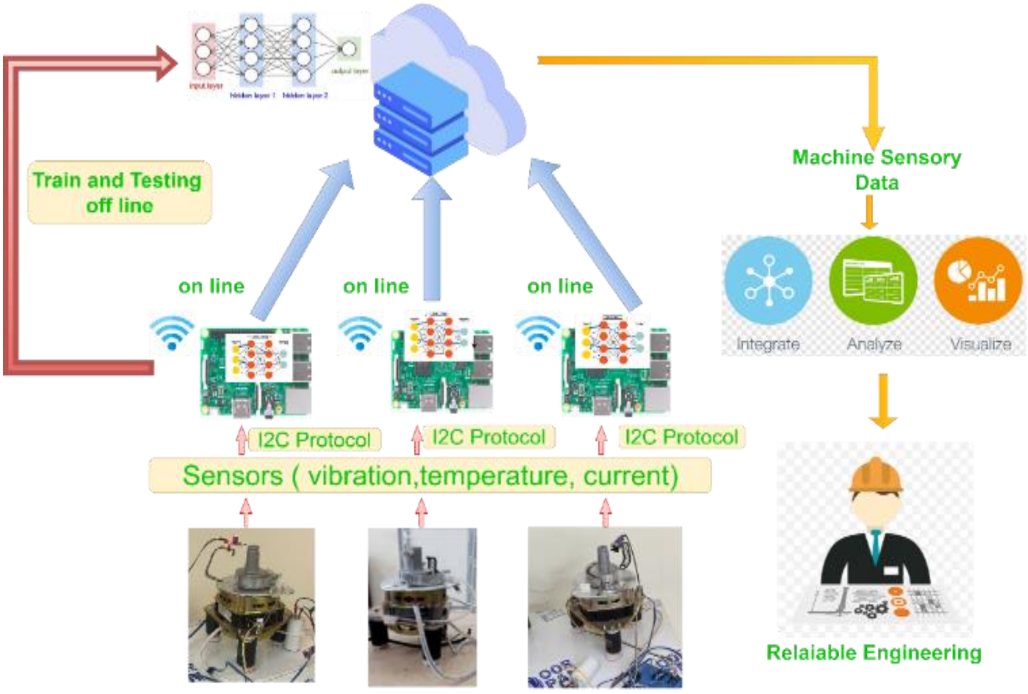

Figure 3. The design and implantation of the proposed system design

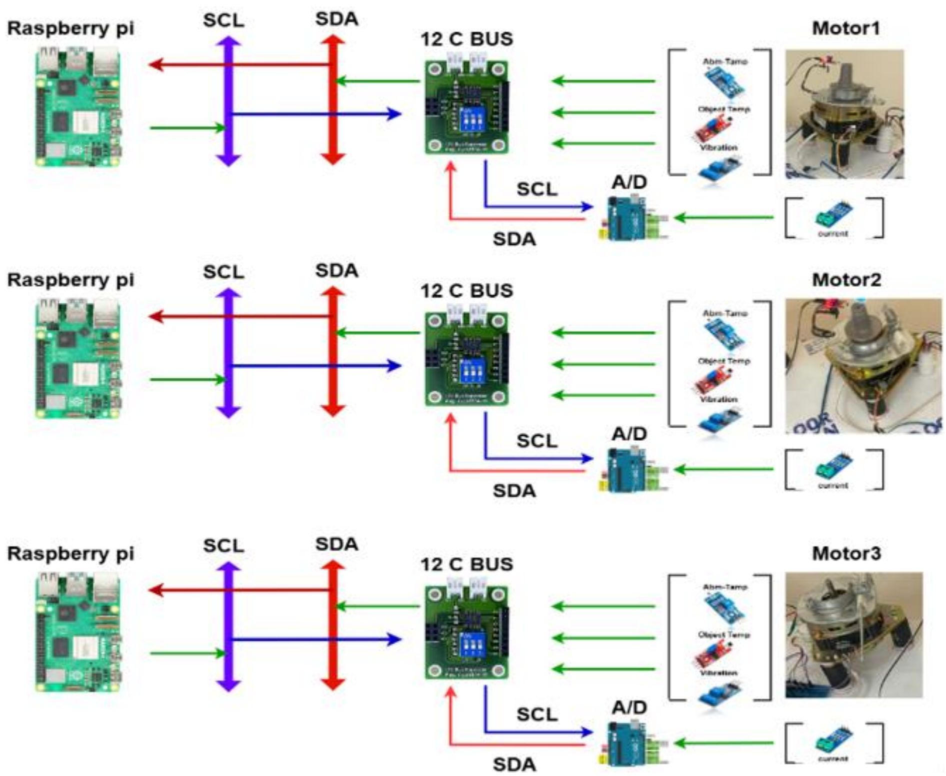

Figure 4. The proposed system architecture when uses the 12C bus protocol

Table 4. Specification of the sensors [20]

Senso	Definition	Measurement Rang	Accuracy	Voltage	Current	Address
ADXL345	Three-axis accelerometer	±2g, ±4g, ±8g, ±16g	0.004g	2.0V-3.6V	0.1mA	0x53
ACS712	Current sensor	±5A, ±20A, ±30A	66mV/A (±5A)	5.0V	10mA (at startup)	Non
MLX90614	Non-contact temperature sensor	-70°C to 380°C	0.02°C	3.6V-5.5V	0.5mA	0x5A, 0x5B
SHT21	Temperature and humidity sensor	0% to 100% humidity, -40°C to 125°C	0.3°C for temperature, 2% for humidity	3.3V-5.5V	0.1mA	0x40
ADC	Device that converts analog signals into digital values.	0-5V	±1% to ±0.1%	0-10V	1 mA	ADS1115

All sensors have the same lines but with different addresses (as described in Table 4). The advantage of assigning a different address for each sensor is to avoid collisions between packets. When designing AC motors, reducing mechanical vibration is an essential consideration. Vibrations may result in adverse effects such as a shortened lifespan, elevated stress, exhaustion, and noise. Significant harm can occur to systems that are vibrating. Measuring mechanical vibrations in operational systems is, therefore, crucial. Obtaining vibration data from all three axes is necessary since these vibrations might appear in axial, radial, and torsional directions. Monitoring the AC motor's current is the second design requirement. The motor is in danger of producing excessive heat if the current rating listed on the nameplate is exceeded. It's imperative to deal with this heat right away to avoid damaging the motor. The third criterion focuses on the need for monitoring motor temperature within the operating ranges. A number of things, such as bushing failure or increased current, can cause temperature rises. Setting the motor's temperature alarm and shutdown limits is crucial to averting such issues. Ignoring this preventative step might cause serious problems for the motor.

Sensor data was gathered every second under both normal and malfunctioning circumstances. An imbalance mass was employed to imitate the required vibration in the motor system brought on by uneven loading. There was an increase in temperature and current when a bigger capacitor was employed to mimic the rise in motor current. This distinction needs to be made at the start of a project by a data.

5. RESULTS AND DISCUSSION

In this work, a DNN model has been used to classify the different types of proposed system failure. Before applying the DNN algorithm, a dataset must be proposed such as normalizing and labeling the different classes as follows:

i) Normalization: Dataset normalization is a curried preprocess step in machine learning. It involves transferring the feature at the dataset to common scales, typically between send 1 or -1. This ensures that no single feature disproportionately influences the model's learning process. By day, the model performance is improved, and the training process is sped up. Also, a fair feature contralto is achieved when features with large ranges are prevented from dominating those with

$$X \text{ scaled} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{4}$$

ii) Dataset labeling: classes are preprocessed to convert categorical data into a numerical formal that machine learning algorithms can understand. It creates new binary columns 0 or 1 for each unique category.

iii) Dataset splitting: Dataset has been split into 70% training and 30% testing, where the training set is used to train the model, which is normally the largest pertain of data. Testing is used to assess the fined performance of the trained model and unseen data. This provides estimating of how well the model generalizes. Figure 5 shows the collected dataset distribution for five types of classes, namely normal, over current, heavy load, stop, and misalignment. As shown in this figure, the class is not uniformly distributed. Figure 6 shows the four types of performance metrics, which are accuracy, loss, precision, and recall. These metrics show the training phase performance based on a training set in the dataset. The training accuracy was 0.9957 while the testing accuracy was 1.00, and the training loss was 0.0029 while the testing loss was 0.0014.

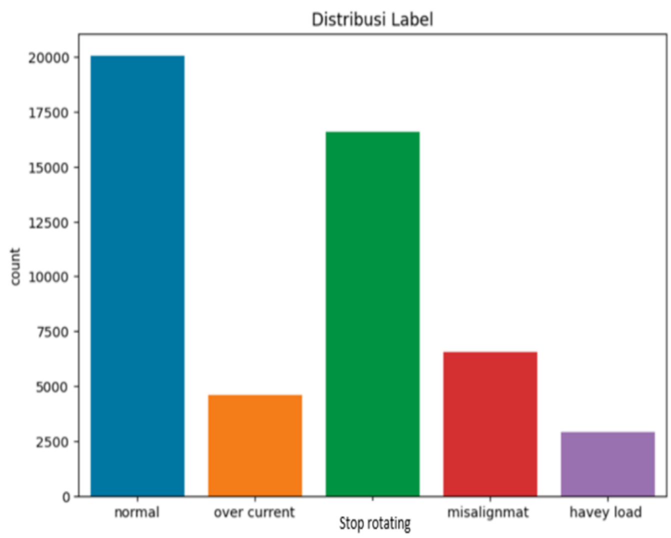

Figure 5. The collected dataset distribution for five types of classes

One performance-measuring method for assessing a classification model's accuracy is a confusion matrix. In supervised learning, where the actual values of the data are known, it is very helpful. Rather than being a 2×2 matrix, a multiclass confusion matrix is an NxN matrix, where N is the number of classes. This type of confusion matrix is an extension of the confusion matrix used to evaluate models that categorize examples into more than two classes. This is especially helpful for issues involving more than two classes. Every component in the matrix The number of times an instance of class i was assigned to class j is denoted by $c_{i,j}$. A complete set of measures for the confusion matrix as a whole may.

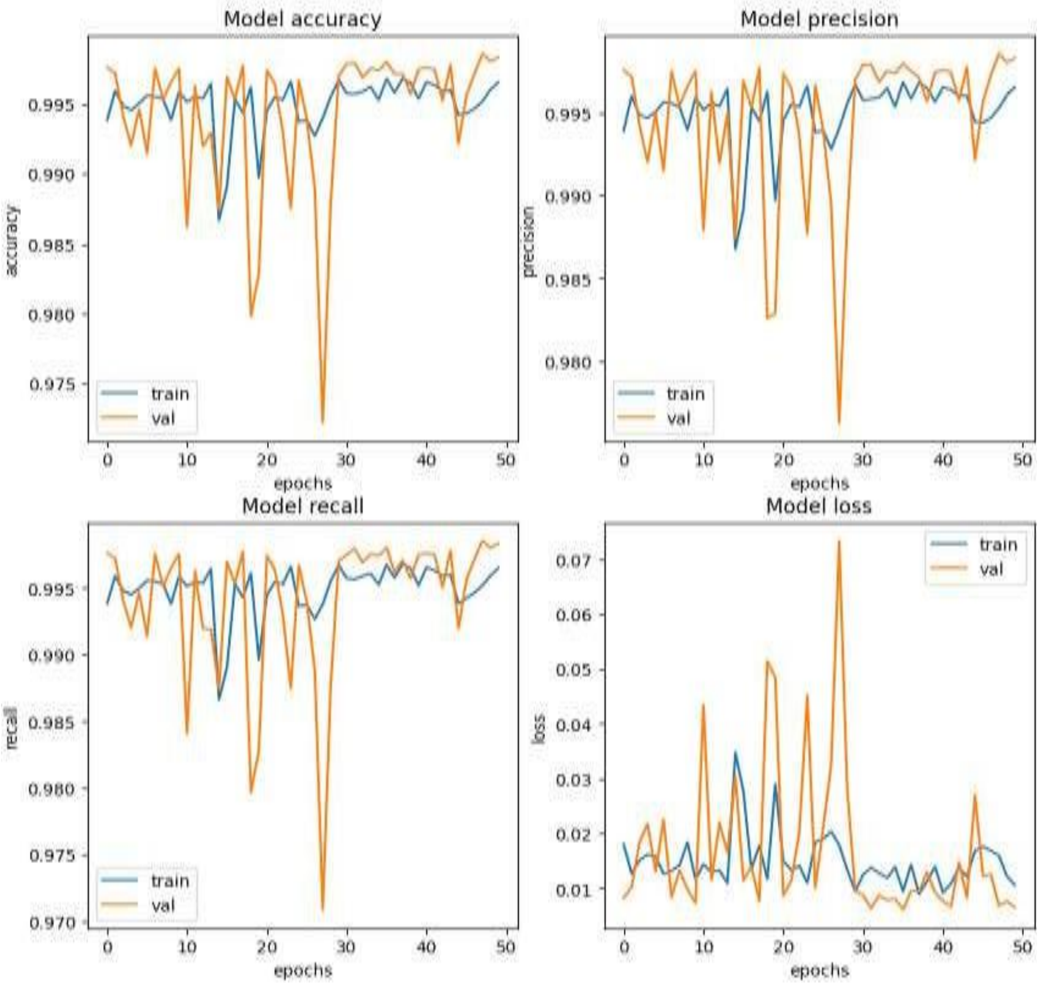

Figure 6. The four types of performance matric

$$Accuracy = \frac{\sum_{i=1}^{N} TP(C_i)}{\sum_{i=1}^{N}\sum_{j=1}^{N} C_{i,j}} \tag{5}$$

$$recall(C_i) = \frac{TP(C_i)}{TP(C_i) + FN(C_i)} \tag{6}$$

$$precision(C_i) = \frac{TP(C_i)}{TP(C_i) + FP(C_i)} \tag{7}$$

$$F1(C_i) = \frac{2 * recall(Ci) * pression(Ci)}{recal(C_i) + recal(C_i)} \tag{8}$$

where,

$TP(C_i)$ is true positive for class$_i$, FN(C_i) is false negative for class$_i$.

Figure 7 shows the multiclass confusion matrix, while Table 5 shows the classification report for each class.

Table 5. Classification report

Class	Precision	Recall	Fl-Score	Support
0	0.98	0.97	0.98	852
1	0.99	0.99	0.99	1993
2	1.00	1.00	1.00	6046
3	1.00	1.00	1.00	1348
4	1.00	1.00	1.00	4967
Accuracy			1.00	15206
Macro avg	0.99	0.99	0.99	15206
Weighted avg	1.00	1.00	1.00	15206

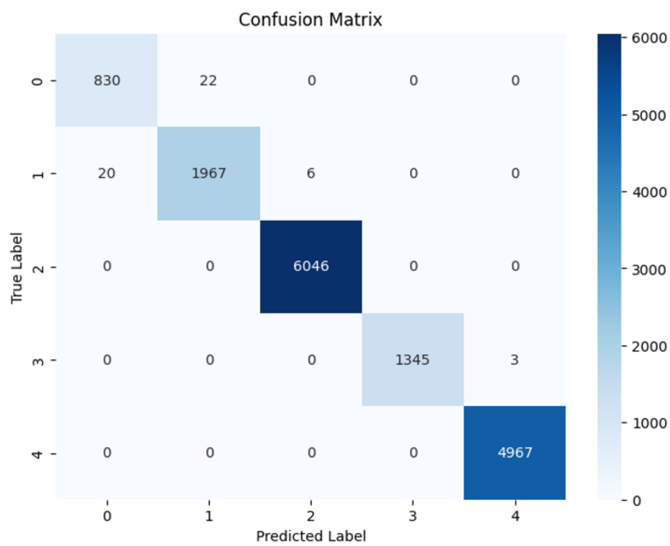

Figure 7. Confusion matrix

6. CONCLUSION

This work effectively built an IIoT-based PdM system to classify problems in industrial AC motors by use of a DNN. By combining three AC motors, four different kinds of sensors: vibration, current, temperature, and ambient temperature. A complete dataset was gathered under both normal and induced fault settings. With a minimum loss of

0.0014, the proposed DNN model showed remarkable performance and 100% accuracy in fault classification throughout testing. Important contributions include the development of a proprietary multiclass dataset, the pragmatic implementation of a cloud-based DNN model for real-time monitoring, and the validation of IIoT-driven PdM as a reliable approach to lower unplanned downtime and maintenance costs.

The findings highlight how well deep learning performs in industrial uses, especially when high-quality sensor data is supporting it. Future research may investigate distributed learning models, including federated or split learning, to improve data privacy and scalability among several machines. Furthermore, improving the dependability and real-time responsiveness of the system might involve extending the dataset to incorporate more varied fault situations and including edge computing for distributed processing. In Industry 4.0 environments, our research opens the path for smarter, data-driven maintenance practices.

REFERENCES

[1] Achouch, M., Dimitrova, M., Ziane, K., Sattarpanah Karganroudi, S., Dhouib, R., Ibrahim, H., Adda, M. (2022). On predictive maintenance in industry 4.0: Overview, models, and challenges. Applied Sciences, 12(16): 8081. https://doi.org/10.3390/app12168081

[2] Khan, W.Z., Rehman, M.H., Zangoti, H.M., Afzal, M.K., Armi, N., Salah, K. (2020). Industrial Internet of Things: Recent advances, enabling technologies and open challenges. Computers & Electrical Engineering, 81: 106522. https://doi.org/10.1016/j.compeleceng.2019.106522

[3] Merkt, O. (2019). On the use of predictive models for improving the quality of industrial maintenance: An analytical literature review of maintenance strategies. In 2019 Federated Conference on Computer Science and Information Systems (FedCSIS), Leipzig, Germany, pp. 693-704. https://doi.org/10.15439/2019f101

[4] Bouabdallaoui, Y., Lafhaj, Z., Yim, P., Ducoulombier, L., Bennadji, B. (2021). Predictive maintenance in building facilities: A machine learning-based approach. Sensors, 21(4): 1044. https://doi.org/10.3390/s21041044

[5] Becker, S., Styp-Rekowski, K., Stoll, O.V.L., Kao, O. (2022). Federated learning for autoencoder-based condition monitoring in the industrial Internet of Things. In 2022 IEEE International Conference on Big Data (Big Data), Osaka, Japan, pp. 5424-5433. https://doi.org/10.1109/BigData55660.2022.10020836

[6] Azari, M.S., Flammini, F., Santini, S., Caporuscio, M. (2023). A systematic literature review on transfer learning for predictive maintenance in industry 4.0. IEEE Access, 11: 12887-12910. https://doi.org/10.1109/ACCESS.2023.3239784

[7] Mohammed, N.A., Abdulateef, O.F., Hamad, A.H., Abdullah, O.I. (2024). Performance analysis of different machine learning algorithms for predictive maintenance. Al-Khwarizmi Engineering Journal, 20(2): 26-38. https://doi.org/10.22153/kej.2024.11.003

[8] Namuduri, S., Narayanan, B.N., Davuluru, V.S.P., Burton, L., Bhansali, S. (2020). Deep learning methods for sensor based predictive maintenance and future perspectives for electrochemical sensors. Journal of The Electrochemical Society, 167(3): 037552. https://doi.org/10.1149/1945-7111/ab67a8

[9] Tsai, Y.H., Chang, D.M., Hsu, T.C. (2022). Edge computing based on federated learning for machine monitoring. Applied Sciences, 12(10): 5178. https://doi.org/10.3390/app12105178

[10] Nasser, A., Al-Khazraji, H. (2022). A hybrid of convolutional neural network and long short-term memory network approach to predictive maintenance. International Journal of Electrical and Computer Engineering, 12(1): 721-730. https://doi.org/10.11591/ijece.v12i1.pp721-730

[11] Drakaki, M., Karnavas, Y.L., Tziafettas, I.A., Linardos, V., Tzionas, P. (2022). Machine learning and deep learning based methods toward industry 4.0 predictive maintenance in induction motors: State of the art survey. Journal of Industrial Engineering and Management, 15(1): 31-57. https://doi.org/10.3926/jiem.3597

[12] Gatta, F., Giampaolo, F., Chiaro, D., Piccialli, F. (2024). Predictive maintenance for offshore oil wells by means of deep learning features extraction. Expert Systems, 41(2): e13128. https://doi.org/10.1111/exsy.13128

[13] Es-Sakali, N., Cherkaoui, M., Mghazli, M.O., Naimi, Z. (2022). Review of predictive maintenance algorithms applied to HVAC systems. Energy Reports, 8: 1003-1012. https://doi.org/10.1016/j.egyr.2022.07.130

[14] Mohammed, N.A., Abdulateef, O.F., Hamad, A.H. (2023). An IoT and machine learning-based predictive maintenance system for electrical motors. Journal Européen des Systèmes Automatisés, 56(4): 651-661. https://doi.org/10.18280/jesa.560414

[15] Zainol, A.M.B., Burhani, N.R.A.B. (2023). Failure prediction for high voltage induction motor using artificial neural network (ANN). In 2023 International Conference on Cyber Management and Engineering (CyMaEn), Bangkok, Thailand, pp. 33-38. https://doi.org/10.1109/CyMaEn57228.2023.10051019

[16] Karwa, R.R., Bamnote, G.R., Dhumale, Y.A., Deshmukh, P.P., Meshram, R.A., Iqbal, S.M. (2024). Predictive maintenance: Machine learning approaches for enhanced equipment reliability. In 2024 2nd DMIHER International Conference on Artificial Intelligence in Healthcare, Education and Industry (IDICAIEI), Wardha, India, pp. 1-6. https://doi.org/10.1109/idicaiei61867.2024.10842946

[17] Al-Naggar, Y.M., Jamil, N., Hassan, M.F., Yusoff, A.R. (2021). Condition monitoring based on IoT for predictive maintenance of CNC machines. Procedia CIRP, 102: 314-318. https://doi.org/10.1016/j.procir.2021.09.054

[18] Singh, D.P., Singh, S.K. (2023). Precision fault prediction in motor bearings with feature selection and deep learning. International Journal of Experimental Research and Review, 32: 398-407. https://doi.org/10.52756/ijerr.2023.v32.035

[19] Karthikeyan, S., Raj, R.A., Cruz, M.V., Chen, L., Vishal, J.A., Rohith, V.S. (2023). A systematic analysis on raspberry Pi prototyping: Uses, challenges, benefits, and drawbacks. IEEE Internet of Things Journal, 10(16): 14397-14417. https://doi.org/10.1109/JIOT.2023.3262942

[20] Ali, M.I., Lai, N.S., Abdulla, R. (2024). Predictive maintenance of rotational machinery using deep learning. International Journal of Electrical and Computer Engineering, 14(1): 1112-1121.

https://doi.org/10.11591/ijece.v14i1.pp1112-1121

[21] Hosny, K.M., Magdi, A., Salah, A., El-Komy, O., Lashin, N.A. (2023). Internet of Things applications using Raspberry-Pi: A survey. International Journal of Electrical & Computer Engineering, 13(1): 902-910. https://doi.org/10.11591/ijece.v13i1.pp902-910

[22] Hussien, H.A., Al-Messri, Z.A. (2023). Synthesis, characterization and evaluation of some meldrum's acid derivatives as lubricant additives. Iraqi Journal of Science, 64(3): 1041-1048. https://doi.org/10.24996/ijs.2023.64.3.1

[23] Abood, R.H., Hamad, A.H. (2024). Multi-label diabetic retinopathy detection using transfer learning based convolutional neural network. Fusion Practice and Applications, 17(2): 279-293. https://doi.org/10.54216/fpa.170221

[24] Obaid, M.H., Hamad, A.H. (2023). Deep learning approach for oil pipeline leakage detection using image-based edge detection techniques. Journal Européen des Systèmes Automatisés, 56(4): 663-673. https://doi.org/10.18280/jesa.560416

[25] Obaid, M.H., Hamad, A.H. (2024). Internet of Things based oil pipeline spill detection system using deep learning and LAB color algorithm. Iraqi Journal for Electrical and Electronic Engineering, 20(1): 137-148. https://doi.org/10.37917/ijeee.20.1.14

[26] Ibraheem, S.S., Hamad, A.H., Jalal, A.S.A. (2018). A secure messaging for Internet of Things protocol based RSA and DNA computing for video surveillance system. In 2018 Third Scientific Conference of Electrical Engineering (SCEE), Baghdad, Iraq, pp. 280-284. https://doi.org/10.1109/SCEE.2018.8684055