IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

Predictive Maintenance of Motors using Machine Learning

Nithish Kanna J L Krishnakumar G, Muhammad Aadhil M, Dr. Ajay V P,
Department of Electronics & Communication Engineering, Kumaraguru College of Technology

Abstract— The suggested predictive maintenance system makes use of sensor readings, operating conditions, and failure incidences from previous motor operation data. Machine learning models are trained on a large dataset, which enables them to identify patterns and correlations suggestive of possible motor breakdowns. A variety of algorithms are used to build a strong prediction model, including ensemble approaches, neural networks, and support vector machines. By continuously analysing real-time data from motors, the predictive maintenance model can identify possible flaws before they become serious failures. Because of this, maintenance teams may plan interventions during scheduled downtime, maximising the use of available resources and reducing unforeseen outages. By extending the lifespan of motors and lowering maintenance costs, the application of this predictive maintenance strategy supports overall sustainability initiatives.

Keywords—predictive maintenance, machine learning models, critical failures

I. INTRODUCTION

In the realm of industrial machinery, the reliable operation of motors is essential for maintaining seamless production processes.[1] The advent of predictive maintenance, powered by machine learning, has revolutionized the approach to ensuring the optimal performance of these critical components. Rather than adhering to traditional reactive maintenance strategies, which often lead to unplanned downtime and increased operational costs, predictive maintenance leverages advanced data analytics to foresee potential motor failures.[2]

This study delves into the realm of predictive maintenance for motors, employing machine learning algorithms to analyze historical data and predict impending faults.[3] By harnessing the power of artificial intelligence, this approach enables proactive intervention, allowing maintenance teams to address issues before they escalate into critical failures.[15] The following exploration outlines the methodology, models, and real-world implications of integrating machine learning into the predictive maintenance framework for motors, with the overarching goal of enhancing operational efficiency and reducing downtime. However, the unpredictability of motor failures poses a significant challenge for industries, often leading to unplanned downtime and increased maintenance costs.[18] In response to this challenge, predictive maintenance has emerged as a proactive strategy to anticipate and address potential equipment failures before they disrupt operations.

IoT predictive maintenance is a maintenance approach that collects and analyses data regarding assets, machinery, or equipment via the Internet of Things. Data regarding the condition of the equipment is gathered by sensors and other devices to identify any problems that might need to be fixed to avoid future failures and needless downtime.[17] You might be wondering what the Internet of Things is, so let us clarify before we go into how IoT predictive maintenance operates. The Internet of Things, also known by its acronym "IoT," is a collection of different hardware, apps, and software that are all linked to one another over the internet. Then, these "things" can communicate useful information with one another to build an extensive and integrated network of effective data.[12]

The amount of data being gathered and analyzed in a variety of industries, including healthcare, transportation, energy, food and beverage, multimedia, the environment, finance, and logistics, has significantly increased because of the recent explosion of smart manufacturing applications, the Internet of things (IoT), and big data.[16] Analyzing different information yields a variety of predictions, including production forecasts, defect identification, and predictive maintenance. Time-series data is one of the most often gathered data kinds in this rapidly emerging Industry 4.0 age. Observations that are sequentially recorded over time are referred to as time-series data.[14]

Time-series data analysis is a highly analyzed preventive measure in the industrial sector, where unanticipated machinery failures can result in extended production downtime and losses. examining and evaluating data to find flaws and risks.[11] Predictive maintenance is an ensemble of activities that detect any abnormal physical condition changes in equipment (signs of failure) to carry out the necessary maintenance procedures to extend the equipment's service life without raising the chance of failure. To enhance, predictive maintenance has been the focus of extensive research in recent years.[13] Using machine learning (ML) approaches with innovative technical concepts to provide improved predictive maintenance outcomes is one of the current revolutionary developments for this subject.[4]

Machine learning (ML) is a branch of artificial intelligence (AI) that uses supervised learning, semi-supervised learning, unsupervised learning, and reinforcement learning to extract meaningful insights from a variety of data, including time-series data. It is often called research that gives robots various tools and methods to carry out jobs correctly and independently without explicit help from people.[5] One area of machine learning called deep learning is capable of extracting data representation. Artificial neural networks (ANN), convolution neural networks (CNN), deep belief networks, recurrent neural networks, and stacked auto-encoders are a few common deep learning techniques.[8]

II. LITERATURE REVIEW

1.Vlasov, A.I. (2018) This paper examines industrial equipment servicing techniques, with an emphasis on predictive maintenance, often called actual maintenance. The following justifies the usage of wireless technology in data processing and collection offered. Depending on the sector of application, the data transmission protocols and building principles for wireless sensor networks are examined to gather statistical data on the condition of the various components of industrial equipment. The study's goal is to demonstrate that employing wireless sensor networks as technical diagnostic tools is feasible from an economic and technological standpoint.

2. **Fordal, J.M. (2023)** The purpose of this article is to investigate how value chain data and maintenance could be used together to improve value chain effectiveness through forecasting. The research process blends theoretical and industrial testing. The study presents a novel concept for a predictive maintenance platform and an artificial neural network (ANN) model that utilises sensor data input. A case study of a company that chose to use the platform is also provided, along with the considerations that led to this decision and its implications. The findings indicate that the platform can be used as a starting point to enable Industry 4.0 and sensor data based predictive maintenance.

3. **TP Carvalho (2019)** The amount of data collected from manufacturing processes has increased exponentially as a result of the development of sensing technology. When data is processed and reviewed, it can be used to extract valuable knowledge and information from industrial processes, machinery, and production systems. In industries, equipment maintenance is essential since it affects the equipment's efficiency and length of operation. Therefore, equipment issues need to be identified and resolved to avoid industrial operations being shut down. Machine learning (ML) approaches have become a viable tool in Predictive Maintenance (PdM) applications to avoid equipment failures in factory floor production lines. However, the performance of PdM applications depends on the choice of ML approach.

4. **Shikhil Nangia (2020)** Industry 4.0 makes it possible for cutting-edge developments in technology, such as machine learning and big data analytics, to blend in with conventional manufacturing procedures to create smart manufacturing. Industrial Internet of things (IIoT) sensors installed on physical assets are used in smart manufacturing approaches to improve production operations.[1] IoT sensors allow smart manufacturing facilities to exchange data autonomously, which can be leveraged to make better informed business decisions. Companies that use smart manufacturing techniques gain a competitive edge because these strategies can increase profit margins, lower maintenance costs, save energy, and produce higher-quality goods. An architecture for IIoT-based predictive maintenance is presented in this paper.

5. **Ovidiu Vermesan (2023)** The use of intelligent edge sensing devices for measuring various parameters (vibration, temperature, etc.) for industrial equipment/motors using artificial intelligence (AI), machine learning (ML), and neural networks (NNs) is being increasingly adopted in industrial predictive maintenance (PdM) applications. Developing and deploying ML algorithms and NNs on edge devices using sensors and microcontroller processing units based on Arm® Cortex®-M cores (e.g., M0, M0+, M3, M4, M7) microcontrollers requires robust AI-based platforms and workflows. This paper highlights the importance of adequately architecting AI workflow for PdM in industrial applications at the edge. New platforms have recently emerged with various degrees of automation and customization for end-to-end development and deployment of edge AI-based algorithms.

6. **AM Mulder (2024)** Myoelectric sensing systems can benefit from these properties by detecting patterns from an array of electrodes to categorise and adapt to a specific individual's muscle movements, enhancing control of robotic systems, or prosthetics. This article discusses a proposed implementation of such a system using edge computing neural networks on an STM32F746 MCU using a TI ADS1198 ADC SPI peripheral. Embedded machine learning has presented the ability to increase the effectiveness of embedded systems by offering a new way to process and adapt to complex data enabling them to detect patterns, anomalies, and repetition in sensor data.

III. MAINTENANCE OPERATION

An effective maintenance programme must be carefully established in order to ensure the longevity and successful functioning of equipment, assets, buildings, and entire companies. The term "maintenance," often known as "technical maintenance," refers to a group of methods and techniques meant to ensure the effective and continued operation of machinery, equipment, and other assets commonly employed in commercial settings.

A. Routine Maintenance

Known by another name, preventive maintenance, this type of maintenance follows a predetermined schedule and typically comprises cleaning, replacing, and checking. In order to avoid any impact on production goals, this is typically done on the weekends or in the intervals between shifts. The two objectives of routine maintenance are to prevent possible problems from emerging by giving regular care and to identify problems early and promptly resolve them.

B. Reactive Maintenance

Reactive maintenance's only objective is to address problems as they arise. Maintenance procedures are necessary when there are equipment problems, setbacks, or failures. When a machine breaks down, it is either repaired or replaced. Cheap, non-critical equipment that has little impact on output or team security is ideal for the run-to-failure approach.

C. Predictive Maintenance

Its main focus is on how to determine when to execute scheduled and corrective maintenance. Its major goal is to predict, through a variety of testing methods, when a machine will start to show noticeable wear and tear. This allows for the scheduling of preventive maintenance ahead of time and the failure of the unit to be avoided, all without compromising production targets.

IV. METHODOLOGY

To perform predictive maintenance the following steps are mentioned below in the flowchart,

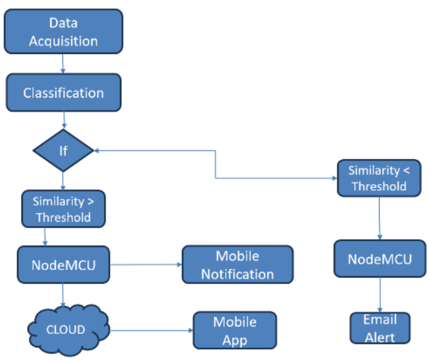


Figure 1: Flow chart

From Figure 1, The data will be acquired from the sensor and sent to the Nano Edge Software and from that software select the suitable Machine Learning model for our data.

Then the library generated from the software imported to the STM cube IDE, where the classification process takes place. After Classification takes place, the data pushes to the cloud continuously with the help of Node MCU.

The instructions given to the controller continuously monitor the training data and check the similarity between the training data and the test data. If the similarity index reaches below the threshold value, then it sends the alert message through email.

V. COMPONENTS AND ITS SPECIFICATIONS

The Components and software used in this project mentioned below,

1. STM Microcontroller
2. MPU 6050
3. Node MCU(ESP8266)
4. STM Cube IDE
5. Nano Edge AI Studio
6. Arduino IDE

1.STM Microcontroller:

The STM32F411RE is a high-performance microcontroller from STMicroelectronics, part of the STM32F4 series. It features an ARM Cortex-M4 core running at up to 100 MHz, making it suitable for a wide range of embedded applications that require both processing power and energy efficiency. The microcontroller is built on a 32-bit RISC architecture, offering enhanced computational capabilities compared to its predecessors.

2.MPU 6050 Sensor:

The MPU-6050 is a motion-tracking device that integrates a 3-axis gyroscope and a 3-axis accelerometer into a single chip. This sensor module provides accurate measurement of acceleration, angular velocity, and orientation in real-time. The gyroscope measures rotational motion around the X, Y, and Z axes, while the accelerometer detects linear acceleration along these axes. It communicates with the host microcontroller via I2C and operates over a wide voltage range with low power consumption. Additionally, it offers configurable sensitivity settings and motion detection interruptions for tailored performance.

3.Node MCU (ESP8266):

The NodeMCU, based on the ESP8266 microcontroller, is a popular development board widely used in IoT (Internet of Things) projects. It offers a convenient and cost-effective platform for prototyping and deploying connected devices and applications. The ESP8266 microcontroller itself is renowned for its built-in Wi-Fi capabilities, making it an ideal choice for projects requiring wireless connectivity.

Overall, the NodeMCU (ESP8266) platform offers a combination of features, affordability, and ease of use that makes it a preferred choice for IoT prototyping and development projects, driving innovation in the rapidly expanding world of connected devices and smart systems.

4.STM Cube IDE:

STMicroelectronics created the STM32CubeIDE integrated development environment (IDE) especially for the purpose of developing software for STM32 microcontrollers. It offers a full suite of tools to make the process of writing, assembling, and troubleshooting STM32 embedded software more efficient.

One of its key features is its seamless integration with the STM32CubeMX configuration and initialization code generator, which simplifies the setup of peripherals and pin assignments, reducing development time and effort. Additionally, STM32CubeIDE offers advanced debugging capabilities, including real-time variable monitoring, breakpoints, and trace features, which aid in the identification and resolution of issues during the development phase.

It supports multiple programming languages such as C and C++, along with a wide range of development boards and hardware debuggers.

5.Nano Edge AI Studio:

With the help of the state-of-the-art platform Nano Edge AI Studio, engineers and developers can now produce creative AI solutions for edge devices. It offers a full range of resources and tools for creating, honing, and deploying machine learning models straight onto tiny, resource-constrained devices like wearable technology, drones, and Internet of Things sensors.

Developers can unlock the potential of edge computing with Nano Edge AI Studio by utilising cutting-edge methods like deep learning and neural networks. This allows for real-time processing and decision-making at the point of data generation. To optimise the performance of AI applications in edge environments and expedite the development process, this platform provides pre-trained models, optimisation methodologies, and an intuitive interface.

6.Arduino IDE:

Manufacturers and experts alike utilise the widely used Arduino Integrated Development Environment (IDE) software platform to programme and create projects using Arduino microcontrollers. For authoring, developing, and uploading code to Arduino boards—which are frequently used in a variety of electronic projects and prototypes—it offers a straightforward but effective interface.

VI. WORKING PRINCIPLE

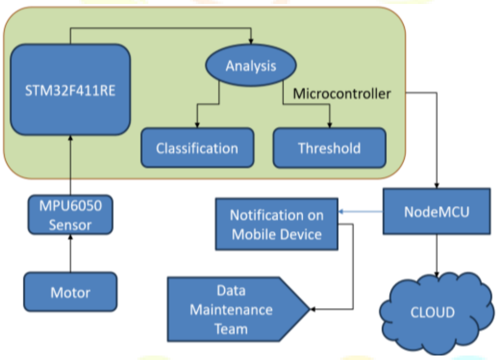


Figure 2: Circuit diagram

We must make connections as per the circuit diagram in Figure 2. We incorporate the sensor over the surface of the motor for acquiring vibrational data. The vibrational data will be processed with the help of Nano Edge Studio.

Figure 3: Hardware implementation

Implementing predictive maintenance using Nano Edge Studio begins with the creation of a project within the platform from Figure 4. This step involves setting up the project environment from Figure 3 and defining the scope of the predictive maintenance application.

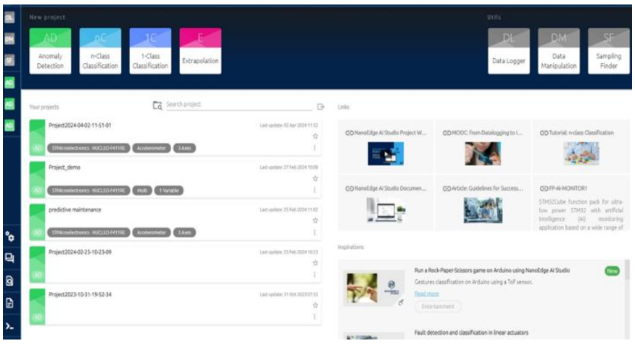


Figure 4: Task Selection

Next, connect the MPU6050 sensor to your development board, ensuring compatibility and proper wiring according to reference materials. This establishes the data acquisition pipeline necessary for capturing sensor data from the motor. Once the data acquisition setup is complete, proceed with preprocessing the raw sensor data within Nano Edge Studio. Implement techniques such as noise removal, data smoothing, and normalization to prepare the data for analysis. This preprocessing stage is crucial for ensuring the accuracy and reliability of subsequent steps in the predictive maintenance process.

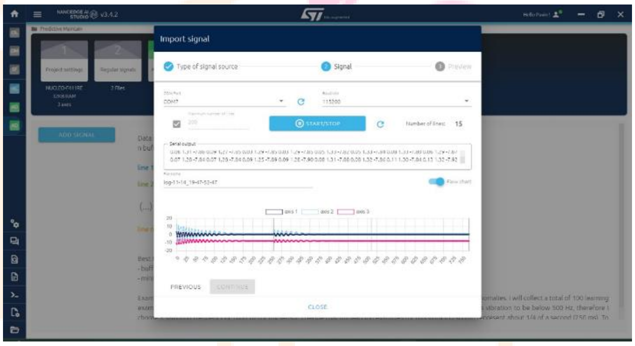


Figure 5: Data Acquisition

Following data preprocessing, focus on feature engineering to extract relevant characteristics from the sensor data. This involves identifying and extracting features that are indicative of the motor's health and performance. Features could include statistical metrics like mean and standard deviation, frequency domain features such as Fast Fourier Transform (FFT), or time-domain characteristics. With the preprocessed data and extracted features in hand, move on to model development within Nano Edge Studio. Choose a suitable machine learning algorithm, such as anomaly detection or classification, based on the specific requirements of your predictive maintenance application. Split the dataset into training and validation sets and train the machine learning model using the training data. Feed the Regular signal with the help of sensor either through USB port or through converting the data into file and train the model from Figure 5. Then execute the following steps based on the type of the project.

After training the model, optimize its parameters to improve performance. This may involve techniques such as hyperparameter tuning or feature selection to enhance the model's predictive accuracy from Figure 6. Once optimized, set thresholds for the model's predictions to trigger maintenance alerts based on domain knowledge or analysis of the training data distribution. With the model trained, optimized, and thresholds set, integrate an alerting mechanism into the predictive maintenance system. Implement mechanisms such as email alerts or dashboard alerts to notify maintenance personnel when potential issues are detected. Finally, deploy the predictive maintenance system in the operational environment, continuously monitoring its performance and refining algorithms as necessary for improved accuracy and reliability.

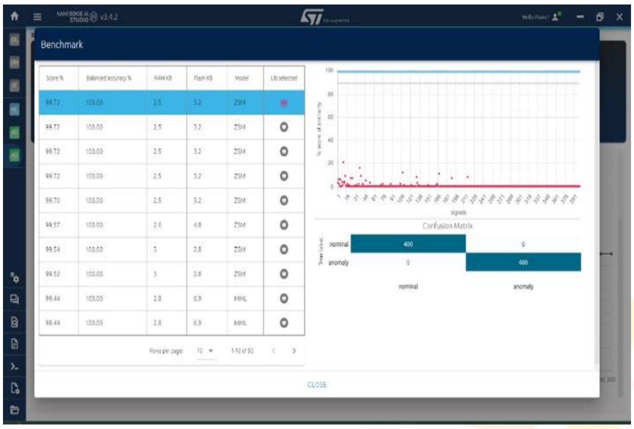


Figure 6: Selection of ML Algorithm

From Nano Edge Studio, we can get a library that contains the training data with suitable machine learning algorithms. We must import the library from the Nano Edge Studio to the STM Cube IDE, which is nothing but the software and is used to give instructions to the STM controller. Once we give instructions to the controller, it will return the similarity index data from the training data. With the help of the Node MCU, the data will be pushed to the cloud. We can monitor the data whenever we want. The data in the cloud continues until the controller is at work.

VII. RESULTS AND DISCUSSIONS

The predictive maintenance of motors has been done with the help of software such as Nano Edge AI Studio, Arduino IDE, and STM Cube IDE. The results of the project are mentioned below.

The below Figure 7 shows the report generated by the Nano Edge AI Studio, which contains the specifications of the signals, such as the input data, cross-validation, performance, and algorithm flowchart. Which gives details about the input signals.

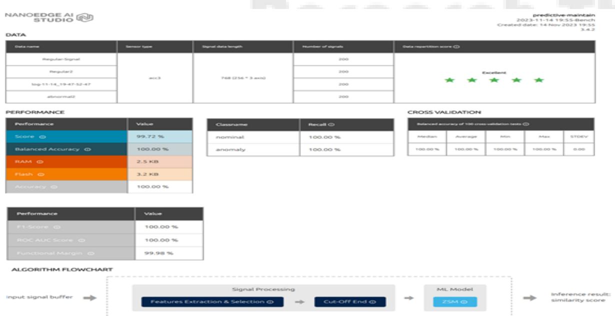


Figure 7: Overall Report of Nano Edge ai studio

Figure 8 shows the similarity index (i.e) the marks will be generated based on how much the test data differs from the trained data. If the similarity between the train and test data is high. Then, the value of similarity index is high. If the similarity between the train and test data is low. Then, the value of similarity index is low. The similarity index values are shown in the command window of STM Cube IDE.

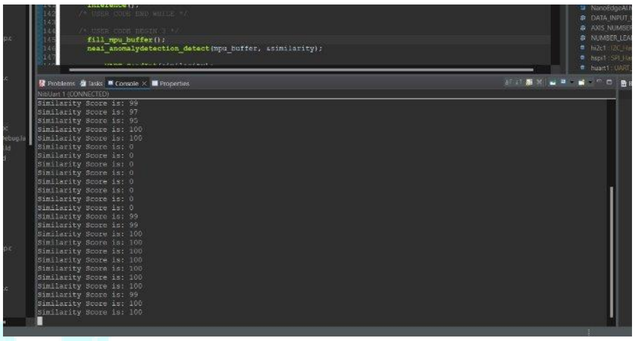
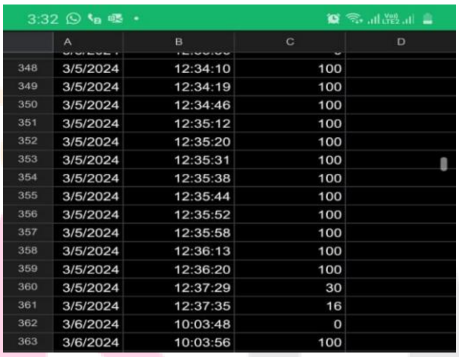


Figure 8: Similarity index value

Figure 9, shows the transmission of vibrational data from an MPU6050 sensor to the Google Cloud Platform (GCP), facilitated by a NodeMCU microcontroller. Through this setup, real-time monitoring of vibrational patterns becomes feasible, enabling predictive maintenance strategies crucial for diverse industrial applications. The hardware setup involves connecting the MPU6050 sensor to the NodeMCU microcontroller using the UART interface. Software configuration involves programming the NodeMCU using the Arduino IDE, integrating necessary libraries such as MPU6050 and Google Cloud IoT Core, and establishing Wi-Fi connectivity for internet access.



	A	B	C	D
348	3/5/2024	12:34:10	100	
349	3/5/2024	12:34:19	100	
350	3/5/2024	12:34:46	100	
351	3/5/2024	12:35:12	100	
352	3/5/2024	12:35:20	100	
353	3/5/2024	12:35:31	100	
354	3/5/2024	12:35:38	100	
355	3/5/2024	12:35:44	100	
356	3/5/2024	12:35:52	100	
357	3/5/2024	12:35:58	100	
358	3/5/2024	12:36:13	100	
359	3/5/2024	12:36:20	100	
360	3/5/2024	12:37:29	30	
361	3/5/2024	12:37:35	16	
362	3/6/2024	10:03:48	0	
363	3/6/2024	10:03:56	100	

Figure 9: Data pushed to the Cloud

Figure 10 shows the graphical depiction of the data mentioned in the preceding figure is displayed in the figure below. To track changes in the dataset from the trained data, the dataset that is transferred to the cloud is visually depicted. The dataset's similarity index value is displayed on the y axis of this graph, while the data and time of data pushes to the cloud are represented on the x axis. An email alert will be issued to the relevant department if the similarity index value drops below the threshold.

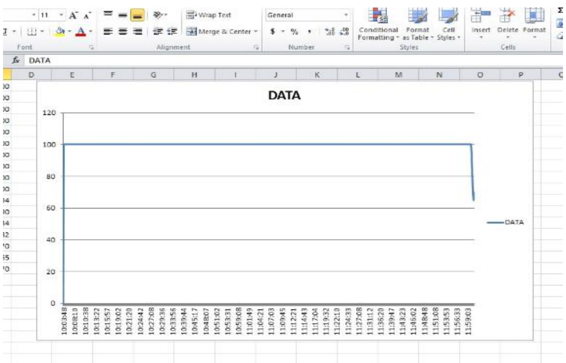


Figure 10: Data Visualization

VIII. CONCLUSION

A major development in industrial maintenance procedures is the application of predictive maintenance approaches for motors, made possible by Nano Edge AI Studio. Through the utilization of edge computing capabilities and machine learning techniques, organizations may anticipate future breakdowns and proactively monitor the performance and health of their motors. By extending the lifespan of essential equipment and lowering maintenance costs and downtime, this improves overall operational efficiency and production. The similarity index data will be monitored anytime with the help of Node MCU, which is used to push data to the cloud.

IX. REFERENCE

1. Kanawaday, A., & Sane, A. (2017, November). Machine learning for predictive maintenance of industrial machines using IoT sensor data. In 2017 8th IEEE international conference on software engineering and service science (ICSESS) (pp. 87-90). IEEE.

2. Kiangala, K. S., & Wang, Z. (2020). An effective predictive maintenance framework for conveyor motors using dual time-series imaging and convolutional neural network in an industry 4.0 environment. Ieee Access, 8, 121033-121049.

3. Patil, S. S., & Gaikwad, J. A. (2013, July). Vibration analysis of electrical rotating machines using FFT: A method of predictive maintenance. In 2013 fourth international conference on computing, communications and networking technologies (ICCCNT) (pp. 1-6). IEEE.

4. Kiangala, K. S., & Wang, Z. (2018). Initiating predictive maintenance for a conveyor motor in a bottling plant using industry 4.0 concepts. The International Journal of Advanced Manufacturing Technology, 97, 3251-3271.

5. Dalzochio, J., Kunst, R., Pignaton, E., Binotto, A., Sanyal, S., Favilla, J., & Barbosa, J. (2020). Machine learning and reasoning for predictive maintenance in Industry 4.0: Current status and challenges. Computers in Industry, 123, 103298.

6. Paolanti, M., Romeo, L., Felicetti, A., Mancini, A., Frontoni, E., & Loncarski, J. (2018, July). Machine learning approach for predictive maintenance in industry 4.0. In 2018 14th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA) (pp. 1-6). IEEE.

7. Lu, B., Durocher, D. B., & Stemper, P. (2009). Predictive maintenance techniques. IEEE Industry Applications Magazine, 15(6), 52-60.

8. Grethler, M., Marinov, M. B., & Klumpp, V. (2021, May). Embedded machine learning for machine condition monitoring. In International Conference on Future Access Enablers of Ubiquitous and Intelligent Infrastructures (pp. 217-228). Cham: Springer International Publishing.

9. Vermesan, O., & Coppola, M. (2023). Edge AI Platforms for Predictive Maintenance in Industrial Applications. In Embedded Artificial Intelligence (pp. 89-104). River Publishers.

10. Nuratch, S. (2017, June). The IIoT devices to cloud gateway design and implementation based on microcontroller for real-time monitoring and control in automation systems. In 2017 12th IEEE Conference on Industrial Electronics and Applications (ICIEA) (pp. 919-923). IEEE.

11. Shetty, R. B. (2018). Predictive maintenance in the IoT era. Prognostics and Health Management of Electronics: Fundamentals, Machine Learning, and the Internet of Things, 589-612.

12. Teoh, Y. K., Gill, S. S., & Parlikad, A. K. (2021). IoT and fog-computing-based predictive maintenance model for effective asset management in Industry 4.0 using machine learning. IEEE Internet of Things Journal, 10(3), 2087-2094.

13. Zonta, T., Da Costa, C. A., da Rosa Righi, R., de Lima, M. J., da Trindade, E. S., & Li, G. P. (2020). Predictive maintenance in the Industry 4.0: A systematic literature review. Computers & Industrial Engineering, 150, 106889.

14. Mobley, R. K. (2002). An introduction to predictive maintenance. Elsevier.

15. 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.

16. Nunes, P., Santos, J., & Rocha, E. (2023). Challenges in predictive maintenance–A review. CIRP Journal of Manufacturing Science and Technology, 40, 53-67.

17. Carnero, M. (2006). An evaluation system of the setting up of predictive maintenance programmes.

Reliability Engineering & System Safety, 91(8), 945-963.

18. Carvalho, T. P., Soares, F. A., Vita, R., Francisco, R. D. P., Basto, J. P., & Alcalá, S. G. (2019). A systematic literature review of machine learning methods applied to predictive maintenance. Computers & Industrial Engineering, 137, 106024.