People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research
University Of Kasdi Merbah Ouargla

Faculty of Information Technology and Communication
Department of Electronics and Telecommunications

Intelligent Predictive Maintenance for induction motor Using Deep Learning in Industrial AC

Presented by

Bougoffa Mohammed Ouassim & Mehani Abderrahmane

President
Dr. Benlamoudi Azzedine

Examiner
Dr. Chaib Ibtissam

Supervisor
Dr. Abdelmadjid Youcefa

Academic Year
2024–2025

Abstract

Predictive maintenance of industrial equipment is necessary for reducing unplanned downtime, reducing repair expenditures and maintaining the competitiveness of a firm. Spontaneous break-downs of induction motors can cost millions in lost revenues. A conventional reactive maintenance approach and time based preventive maintenance techniques can be too late to be useful or wasteful of resources, whereas today's data-driven predictive maintenance PdM can predict faults before they happen. We have created and evaluated three deep-learning architectures for PdM of three phase induction motors a pure temporal long short-term memory (LSTM) network, a fuzzy-logic adaptive network based fuzzy inference system (ANFIS) model, and a new end-to-end hybrid of ANFIS and LSTM.

We built six-sensor data streams (three-phase currents, three-phase voltages, vibration, temperature and speed) into historical and current time sliding windows, and marked four fault classes: normal, bearing, mechanical, and winding. After separating train/test partitions in a stratified manner, we trained and evaluated based on accuracy, precision, recall, F1-score, and confusion matrix. The hybrid ANFIS+LSTM model combines interpretive fuzzy inference and rapid temporal feature learning in such a way as to both rapidly converge and robustly generalize, reached near perfect fault detection performance. By combining ANFIS and LSTM in an end-to-end design, we were able to combine fast temporal feature learning with interpretable fuzzy rules. The model achieved 98% convergence in fault detection, with rapid convergence and consistent accuracy across all types.

Keywords: Predictive Maintenance PDM, Induction Motor, ANFIS, LSTM, Deep Learning, Fault Detection, Fuzzy Logic, Preventive Maintenance (PM), Reactive maintenance (RM) , Machine Learning.

Contents

List of Figures

List of Tables

GENERAL

INTRODUCTION

1 Introduction

Maintenance is a very important function when it comes to industry since it has a direct impact on cost, reliability, and the competitive advantage obtainable. Unplanned failures of equipment can cause severe financial and reputational damage such as happened with Amazon's 49-minute outage in 2013 which resulted in a $4 million cost to the company. Data centers lose an estimated $138,000 hour when down. In addition, regular maintenance costs can take a significant share of operating budgets or overall costs, such as the category of offshore wind which incurs between 20% and 35% of revenue on maintenance with oil and gas industries reaching upwards of 70% of production costs so it is critical that companies establish a very clearly defined maintenance strategy which can minimize unforeseen downtime, enhance reliability, and provide costs of maintenance across the overall cost of operations.

The development of modern methods (the Internet of Things, sensor technology, artificial intelligence, and others) reflects a shift in maintenance strategy from reactive maintenance (RM) to preventive maintenance (PM) and then to predictive maintenance (PdM). This means that RM is implemented only to restore operational equipment. It is a measure of the physical condition of equipment after a failure, and therefore tends to result in high repair costs due to reactive maintenance and low response time. Periodic maintenance (PM), on the other hand, is implemented according to timed or continuous iterations of the process, leading to disaster (because the latter can lead to unnecessary maintenance costs and high prevention costs). PdM seeks to strike an ideal balance between the two. It is based on an electronic assessment of "condition" and can assist in timely interventions before failure. PdM allows maintenance frequency to be kept as low as possible, avoiding unscheduled routine maintenance without incurring additional maintenance management costs.[1]

Figure 1: Predictive Maintenance (PdM) Stages.

2 Problematic

Industrial AC systems are prone to multiple challenges, which have anadverse effect on operational efficiency and cost-effectiveness. Nobody wants unplanned downtime caused by unexpected failures, but with traditional maintenance,the process falls either toward "too much" preventive maintenance that slows down production or "too little," with the reactive repair needed after something breaks down. Additionally, defective parts in thesesystems can lead to technology misfires, where energy consumption soars but efficiency drops, placing double the load on operational bottom lines. Moreover, insufficient fault diagnosis not only endangers safety, but also threatens the safety of expensive equipment, while the largeamount of sensor data generated makes it difficult for traditional methods to extract meaningful insights. Intelligent predictive maintenance using deep learning addresses these issues by continuously analyzing real-time data, accurately forecasting potential failures, and optimizing maintenance schedules, thereby enhancing reliability, reducing downtime, and improving energy efficiency.

3 Motivations

Intelligent predictive maintenance using deep learning techniques has gained momentum as an emerging technology in industrial AC conditioning systems, driven by the need to improve equipment performance and reduce operating costs. This new approach has encouraged companies that suffer from unplanned downtime, resulting in lost revenue and production, as well as negative brand impact. Furthermore, you can adopt a more proactive, data-driven approach to achieving your goals, moving away from traditional reactive or preventive maintenance strategies, to determine the most appropriate approach to improve equipment performance, reduce energy consumption, and extend its lifespan. Deep learning integration at the plant level also addresses the challenge of processing massive amounts of sensor data to provide more accurate failure detection with timely interventions that ensure safer and more sustainable operations.

4 Contributions

This proposed thesis presents a technique that uses deep learning methods to classify and detect faults in three-phase motors used in industrial processes in AC systems. Using state-of-the-art sensor data acquisition techniques, data signal processing, and the latest developments in deep learning for near-real-time fault prediction, this methodology not only identifies faults in operating systems but also provides a rapid method for monitoring them. This methodology leverages this information to provide an accurate and stable fault detection solution, improving system reliability, reducing unplanned downtime, and simplifying maintenance schedules. This research represents a significant advance over traditional maintenance strategies, paving the way for more efficient and cost-effective industrial operations.

To achieve our objectives, we employed the following methodology:

1- LSTM-Based Deep Learning: This part of the methodology uses Long Short-Term Memory (LSTM) networks to analyze time-series sensor data collected from industrial AC systems. As LSTM networks are good for capturing temporal dependencies, the model learned to recognize weak patterns and early footprints of probable faults over time. After analyzing the data using deep learning the system can spot anomalies that traditional analytics would overlook.

2- Diagnosis by ANFIS: Diagnosis by ANFIS So Fuzzy rule base boosted by Neural Networks (NN) combine the advantage of both Fuzzy rule base and a Neural Network. This method addresses the inherent uncertainty and non-linearities in sensor data which allows the system to interpret complex input patterns and provide an accurate fault diagnosis. This makes ANFIS a more accurate and adaptive detection process.

3- Integration of ANFIS and LSTM: By knowing the two types of techniques have distinct merits and provide more complementary benefits, the method adopts synthesizing the output from LSTM-based deep learning model and ANFIS diagnosis. The integration level logic of combining the strong temporal capabilities of LSTM with the Fuzzification and rule-based features of ANFIS enables the high accuracy and reliability of the fault detection framework.Through this fusion process, it confirmed that the final prediction combines the advantages of both techniques and thus be an effective and reliable solution for detecting faults on AC systems using induction motors.

Chapter 2

Work background

1 Introduction

In the 21st century, smart manufacturing, or Industry 4.0, has been developed through the convergence of cyber-physical systems and advanced computing infrastructures such as cloud computing, big data analytics, other Internet of Things (IoT) platforms or frameworks, and artificial intelligence (AI).

This leverages systems capable of responding in real time to dynamic, real-time conditions on the factory floor and/or supply chain, as well as changes in customer requirements across the manufacturing ecosystem. However, many of these interactions involve human resources, physical assets, and intangible assets, and processes require massive amounts of data generated throughout their entire life cycle. This data, when processed through pre-trained AI algorithms, significantly impacts decision-making, contributes to proactive strategies, and defines processes for production line optimization, leading to business optimization. Various AI technologies have been widely applied to extract valuable insights from multivariate data, reduce operational complexity, and identify critical relationships between processes. This enables manufacturers to forecast production trends, reduce variances, and improve overall productivity and product quality.

AI also plays a pivotal role in predictive maintenance, continuously monitoring equipment performance and analyzing sensor data to identify early indicators of potential failures. This enables timely maintenance interventions that reduce unplanned downtime, minimize repair costs, and extend the life of critical machinery. By leveraging historical data and knowledge gained from previous operational decisions, AI not only complements existing knowledge but also delivers actionable insights that advance traditional production line management to create smart, adaptable, and highly efficient environments.[2]

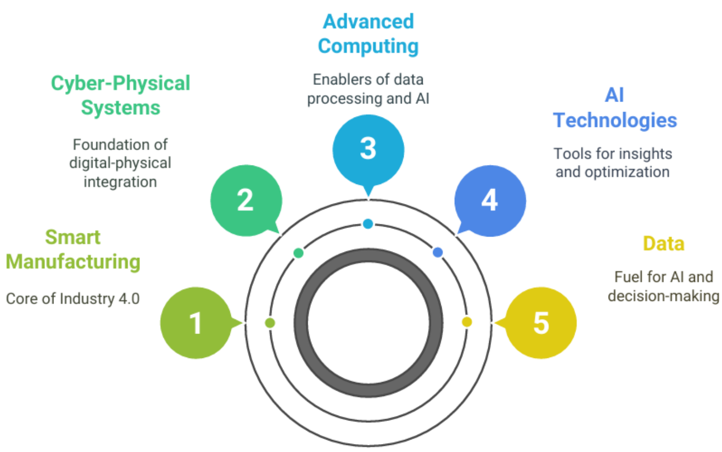

Figure 2: Smart Manufacturing Ecosystem.

2 Based on PHM Systems for Predictive Maintenance

Predictive maintenance (PdM) uses prediction and health management (PHM) systems to repurpose raw data from sensors into actionable information for continuous equipment health monitoring and maintenance. By periodically measuring, collecting, and analyzing operational information, as well as physical parameters such as vibration, temperature, pressure, and flow rate, PHM systems help accelerate fault detection and condition-based estimation of an asset's remaining useful life (RUL).

This comprehensive data set is analyzed by machine learning algorithms and statistical models to evaluate and predict failure patterns for scheduling and planning maintenance and repair activities. This combined approach helps reduce unplanned downtime, lower maintenance costs, and enhance overall safety and reliability by intervening with maintenance support before significant issues arise. Ultimately, the concept of predictive maintenance (PdM) is essential to any rational combination of smart sensors, network based data collection, and cloud-based analytics integrated into a PHM approach.[3]

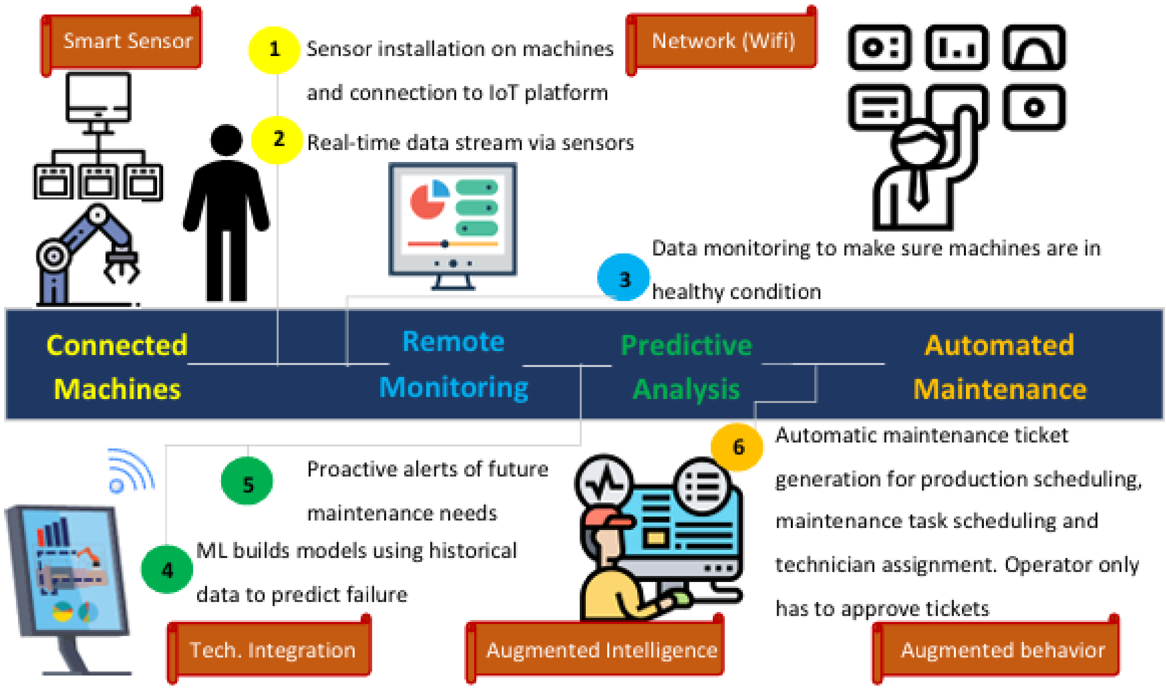

Figure 3: PdM process and technologies to drive PdM.

3 The Role of Artificial Intelligence in Predictive Maintenance

Predictive maintenance (PdM) using artificial intelligence (AI) involves assembling intelligent algorithms to monitor, analyze, and predict the condition of machines using data from sensors. First, on machine tools during operation, external sensors (such as speed, vibration, temperature, and tension) are installed to form raw analogy signals that reflect the machine's operating condition. These signals are first filtered in time or frequency domains to capture condition information, using noise and all unwanted frequency components.

These "features" serve as "inputs" for AI models that can be taught using different learning methods supervised, unsupervised, or reinforcement. In supervised learning, models learn from labeled data (e.g., data with normal or faulty states), while unsupervised learning identifies patterns within data without relying on predefined labels. Reinforcement learning leverages reward-based learning to improve decision-making. Simply put, once trained, the AI model, called "Oncetrained," can estimate the current or future health status of machines based on real-time data, enabling early detection of faults, reducing unexpected downtime, and optimizing maintenance strategies. Thus, AI plays a vital role in revolutionizing traditional maintenance methods toward intelligent, data-driven systems.[4]

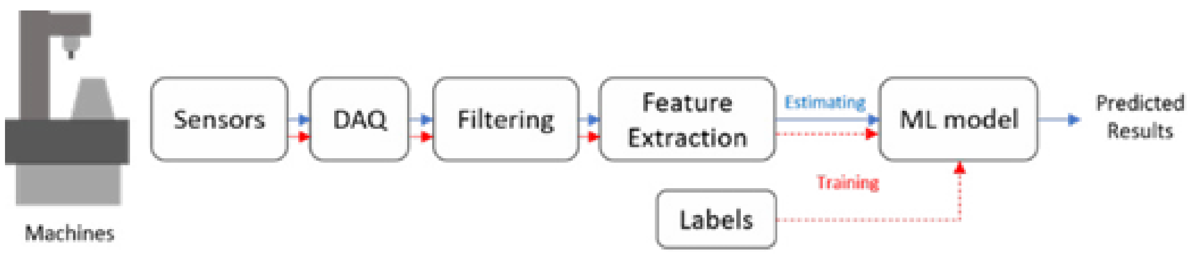

Figure 4: The procedure of PdM using supervised learning algorithms.

4 Machine Learning

Machine learning has become increasingly important for organizations hoping to use their data to make better decisions and increase their efficiency. Machine learning provides the ability to create systems that read data and learn patterns and behaviors, compared to traditional programming that relies on explicitly coded instructions. Various forms of algorithms are used in this data-driven approach, where the process is iterated multiple times to learn and improve for increased efficiency.

Machine learning is a type of artificial intelligence that uses data to learn and improve performance over time. These "models" embody the "relationships" and "knowledge" extracted from data, enabling systems to complete tasks such as classification, regression, anomaly detection, and prediction. As a result, machine learning plays a pivotal role in the development of intelligent systems across a wide range of fields, from business analytics to industrial automation and predictive maintenance.[5]

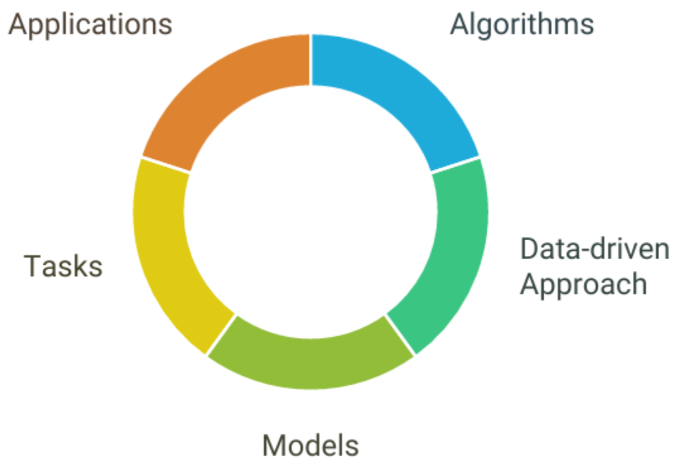

Figure 5: Machine Learning Overview.

4.1 Machine Learning-Based Fault Detection and Diagnosis (FD/D) Methods for Induction Motors (IMs)

Compared to other types of electric motors, induction motors (IMs) are the most widely used in industrial applications due to their robustness, simplicity, low cost, and high-performance characteristics. However, as with all electromechanical systems, induction motors can be subject to various types of failures, such as electrical failures (e.g., stator and rotor winding failures) or mechanical failures (e.g., bearing failures, broken rotor bars, and eccentricity).

Timely and accurate detection of these problems (FD/D) has become critical to prevent unexpected breakdowns, reduce maintenance costs, and ensure operational efficiency. Traditional fault diagnosis methods often rely on manual inspection or rule-based systems, which may not perform efficiently with the complexity and scale of contemporary manufacturing environments.To address these limitations, machine learning (ML) techniques have been proposed as effective tools for accurate analysis and diagnosis (FD/D) of manufacturing machines. It is a gap-based approach to learning patterns associated with normal and faulty engine conditions from large amounts of sensor data, such as vibration signals and motor current measurements. Data collected from faulty structures can be processed in real time using algorithms such as support vector machines (SVMs), artificial neural networks (ANNs), k-nearest neighbors (kNNs), decision trees, and clustering methods. These algorithms can be trained to detect and classify faults with high accuracy.

The convergence of artificial intelligence and condition-based maintenance (CBM) strategies, particularly with regard to motor current signature analysis (MCSA), enables real-time, non-intrusive monitoring of the health of information management systems. As a result, machine learning-based FD/D methods are now an essential component of intelligent maintenance systems in industry, and investigations into FD/D are informing the industry 4.0 of predictive maintenance strategies that address time with greater reliability and performance for industrial activities.[6]

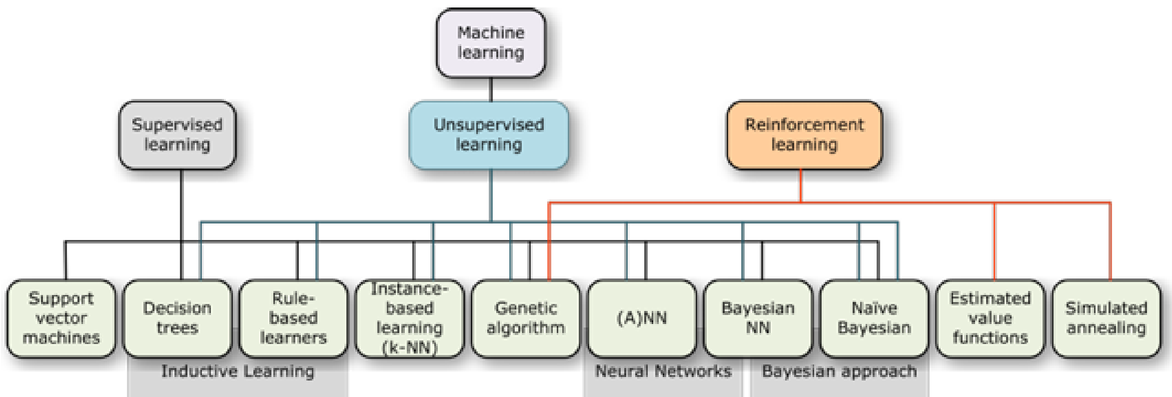

Figure 6: Classification of main ML techniques.

4.2 Fault Detection and Diagnosis (FD/D) Based on Multi-Agent Systems (MAS) Employing Machine Learning Classifiers

Modern smart manufacturing has revolutionized the design and implementation (FD/D) process by integrating multi-agent systems (MAS) and machine learning (ML) classifiers. MASs consist of autonomous intelligent agents capable of collaboration, communication, and real-time decentralized decision-making, providing a flexible and modular approach to managing complex systems. In predictive maintenance (PdM), MASs monitor and diagnose failures decentrally by assigning each agent to a specific machine component or failure condition.

These agents are often trained using machine learning classifiers such as artificial neural networks (ANNs), support vector machines (SVMs), k-nearest neighbors (kNNs), and multi-layer perceptrons (MLPs) to identify and estimate failure conditions from sensor data. For example, in various types of actuators, agents can read rotor and stator conditions using data collected from motor current signature analysis (MCSA) using fast Fourier transform (FFT) to detect performance degradation caused by rotor bar failure or stator short circuits.

This agent also solves some of the problems in this area by heuristically aggregating local agent decisions, even in the presence of inconsistencies, and ensures fault diagnosis and tolerance. The use of MAS within machine learning classifiers increases the adaptability and reliability of FD/D systems, making it an ideal candidate for use in the dynamic, data-intensive manufacturing environments that characterize Industry 4.0.

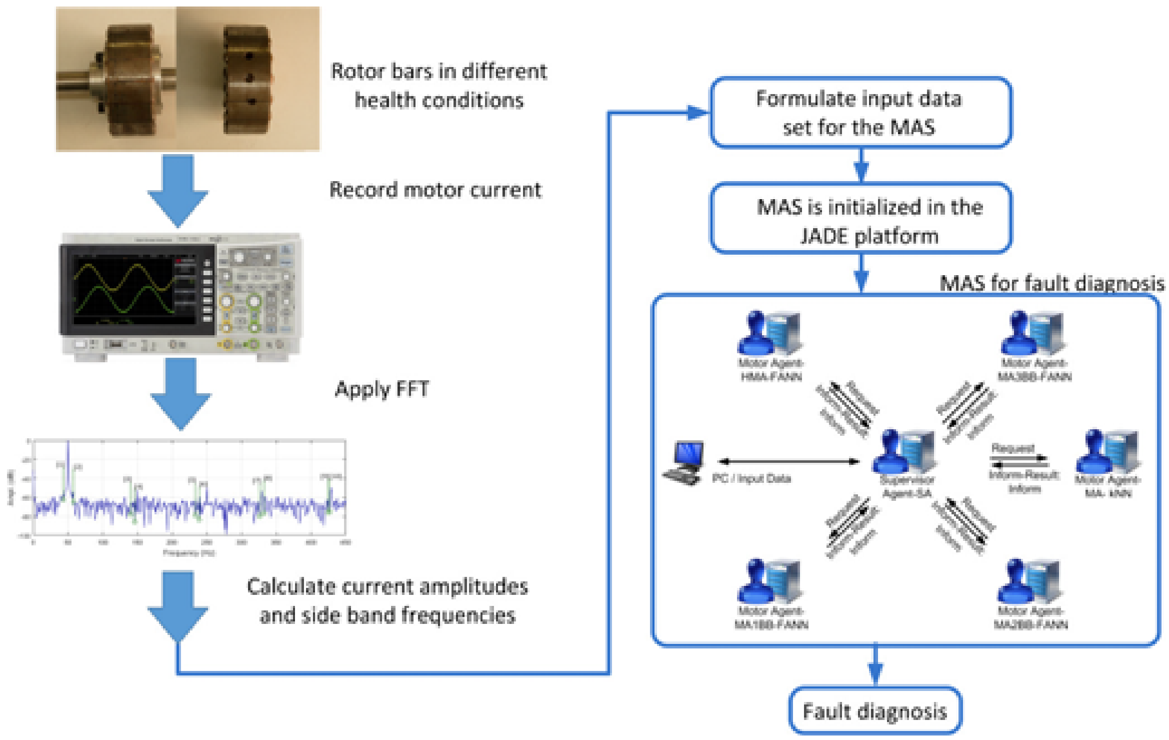

Figure 7: MAS based FD/D decision support framework.[6]

4.3 Fault Detection and Diagnosis (FD/D) of Induction Motors Based on Neural Networks (NNs)

FD/D technology for industrial machinery features the use of neural networks as a highly effective and powerful framework that promises more reliable operation of such critical industrial machines. While both categories of machines are highly valued components of modern society due to their robustness and efficiency, they are unfortunately prone to failures, including inter-cycle short circuits, bearing failures, and broken rotor bars, all of which can cause severe operational disruption.

Over time, the field of neural networks (NNs) has been introduced to computational modeling, particularly for structures such as multilayer perceptrons (MLPs) trained through backpropagation, where they have been widely used to approximate complex, nonlinear relationships within high-dimensional sensor data.

These ANN-based methods often utilize sophisticated signal processing techniques, such as the discrete wavelet transform (DWT) or fast Fourier transform (FFT), to extract key features from current, vibration, or other relevant signals, enabling accurate classification of various failure conditions. This inductive result of combining neural network architecture with advanced feature extraction not only increases the measured fault detection accuracy by 95%, but also provides real-time predictive maintenance strategies, resulting in reduced downtime and maintenance costs. Thus, this latest work begins the application of NNs in FD/D IMs and represents a significant step toward industrial diagnostic systems that meet the needs of smart manufacturing and Industry 4.0.

4.3.1 Artificial Neural Network (ANN)

Artificial neural networks are capable of genetically modeling complex nonlinear relationships, they are widely used in mechanical fault diagnosis. Inspired by the hydration of brain neurons, artificial neural networks learn from data, adapt to changes, and automatically extract important features. Backpropagation multilayer perceptrons (MLPs) are widely used in training and pattern recognition. Artificial neural networks are flexible in handling noisy data and can provide accurate fault detection.

However, they typically require extensive, high-quality training data and may not generalize well to new domains without retraining. Finally, system failures can also lead to more complex problems; we used artificial neural network methods because of their power, ease of application, and flexibility in handling such problems. The neural network's output "X" and The number of hidden layers in the network "Nh" can be expressed as:

$$X = f\left(\sum_{i=1}^{n} a_i w_i + b\right) \tag{1}$$

$$N_h = \frac{N_s}{\alpha(N_i + N_o)} \tag{2}$$

where:

- $w_i = Weight$
- $b = bias$

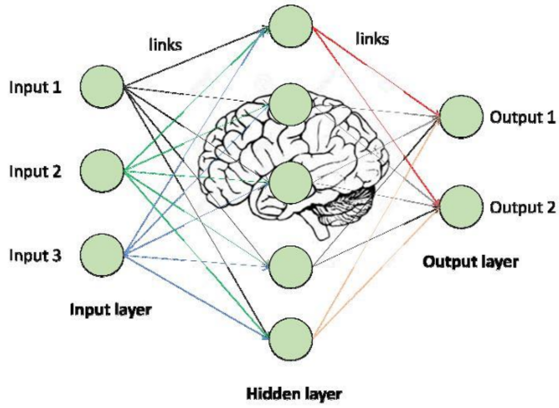

Figure 8: Architecture of neural network.

4.3.2 Support Vector Machine (SVM)

Support vector machines (SVMs) are powerful supervised learning techniques used in classification and regression problems, and are particularly useful in high-dimensional domains with sparse data. Rooted in statistical learning theory and structural risk minimization, SVMs work by identifying the optimal hyperplane that optimally separates data classes. The optimization problem forms the core of the SVM algorithm, solving a quadratic function that seeks to minimize classification error and maximize the margin between data points.

SVM classifiers have proven effective in many applications, particularly in machine fault diagnosis due to their robustness, ability to handle small datasets, and nonlinear classification. Multi-class fault classification and single fault detection strategies. Common strategies in single fault detection applications, or single fault detection strategies, are easy to use for multi-class fault classification and typically outperform other methods in comparative studies.[7]

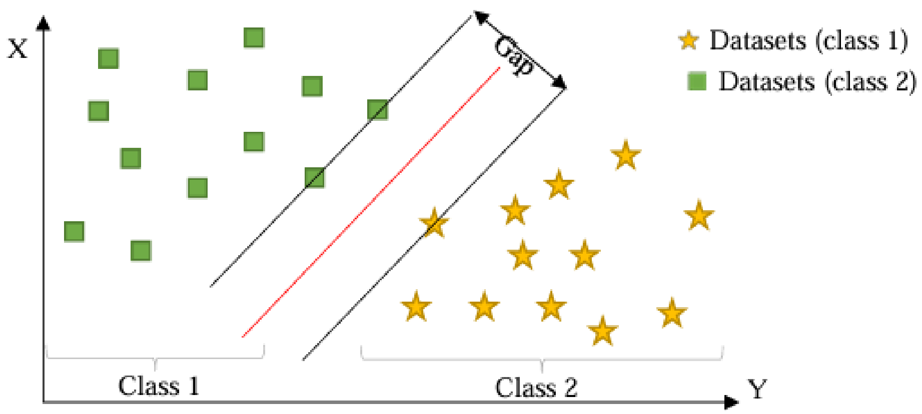

Figure 9: Support vector machine algorithm.

4.4 FD/D Based on Other ML Techniques

In addition to traditional artificial neural networks, other machine learning approaches have been used for fault detection and diagnosis (FD/D) in induction motors. These strategies employ techniques such as unsupervised learning and clustering to detect trends and anomalies in complex, high-dimensional sensor data without the need for large, annotated datasets. For example, Skowron et al. (2019) proposed the use of a self-organizing Kohonen network with ISCA analysis of stator currents for fault detection in transformer-fed induction motors.

This method was also able to efficiently cluster similar fault conditions, resulting in high accuracy for rotor faults (93–95%) and no fault conditions (95%), with somewhat lower performance for stator faults (74%) and mixed faults (72%). The relative success of such contextual forms of machine learning techniques reinforces the view that they represent robust and robust solutions to industrial diagnostic problems when a full evaluation of a principle-based FD/D model approaches true computational benchmarks.

4.5 FD/D Addressing Multiple ML Techniques

By applying various machine learning (ML) techniques, fault detection and diagnosis (FD/D) in induction motors has maintained its advancements, surpassing traditional neural networks until October 2023. Several intelligent classifiers, such as naive Bayes, k-nearest neighbors (k-NN), support vector machines (SVM/SMO), multilayer perceptrons (MLP), decision trees (C4.5), and rule-based methods such as RIPPER, address the complex problems associated with fault diagnosis in induction motors.

These methods have demonstrated very high classification accuracies, typically exceeding 90% for various fault types (stator faults, broken rotor bars, and range bearing defects). For example, the accuracy of kNN and MLP methods has fallen to near-perfect accuracy for some small motor faults, while SVM and composite approaches, such as RandomForest, have generally achieved good results on multi-objective problems. Several approaches, such as Fuzzy ARTMAP and Bagging, were also discussed, each outperforming the other based on the type of fault and operating conditions. This approach does not focus on hiding system defects, which enhances system efficiency, paving the way for more proactive maintenance and improved system reliability.

5 Deep learning

Deep learning (DL) has revolutionized fault diagnosis, enabling systems to automatically learn hierarchical, rotational, scale, and translational invariant features from raw sensor data, enabling them to learn features without human intervention. Using architectures such as convolutional neural networks (CNNs), recurrent neural networks (RNNs), and autoencoders (AEs), DL-based models can capture the temporal and spatial dependencies present in complex datasets a feature essential for effectively learning the subtle features that play a role in separating different fault states.

This enables DL-based methods to address high-dimensional and heterogeneous data fusion problems and transform complex information into supervised multi-label classification tasks. As a result, DL has become an effective method for predictive maintenance and condition monitoring, delivering advanced performance and improving the reliability and efficiency of diagnostic systems in industry.[6]

5.1 Convolutional Neural Networks (CNNs)

Convolutional neural networks are an important class of deep learning architectures that have revolutionized pattern recognition and image processing tasks and are increasingly being applied to fault diagnosis. Most CNNs consist of alternating convolutional and pooling layers, followed by fully connected layers.Their convolutional layers use trainable filters, or kernels, that slide over the input data, generating feature maps that highlight different data values in a translation-invariant manner.

These maps allow the network to learn hierarchical representations, with each subsequent convolutional layer responsible for more abstract features. Pooling layers then perform down-sampling of the feature set, resulting in a low-dimensional data representation while preserving the latest information regardless of its spatial location. This "hierarchical extraction" of features is finally flattened into a one-dimensional vector and fed to fully connected layers, culminating in a "SoftMax" layer that performs probabilistic classification. Figure 7 shows a CNN architecture.

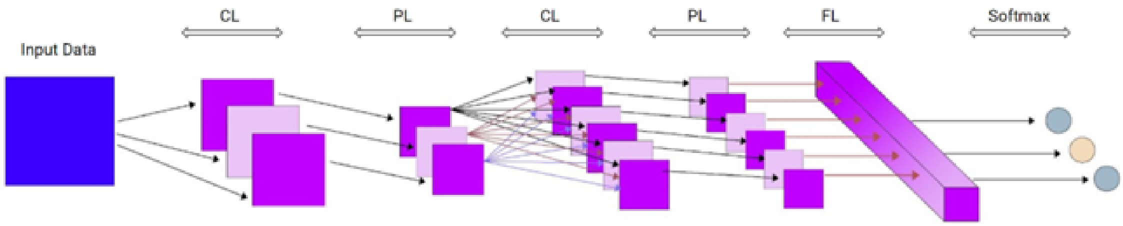

Figure 10: A CNN architecture consisting of two pairs of alternating convolution and pooling layers followed by a fully connected layer and a softmax layer.

5.2 Recurrent neural networks (RNNs)

Recurrent neural networks (RNNs) are defined for taking sequential data as input, where the hidden state is a function of the previous hidden state via recurrent connections. They are ideal for time-dependent application tasks, such as time series tasks like engine fault diagnosis. On the other hand, RNNs can be trained, but this can be difficult due to problems with vanishing gradients and exploding gradients. Gradient clipping and weight regularization can mitigate these problems. RNNs contain typical components that optimize specific quantities over time.

Long short-term memory (LSTM) modules, a specific class of recurrent neural networks, can efficiently remember long-term dependencies, thereby improving performance. LSTMs are very effective at keeping track of fault evolution over time. Figure 8 shows an RNN.

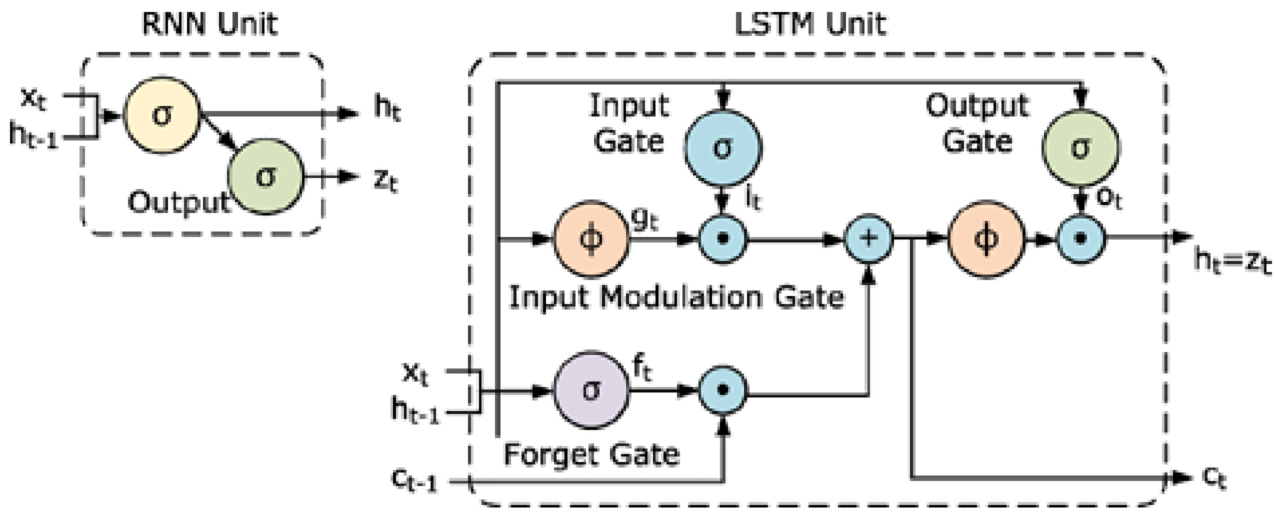

Figure 11: The RNN unit consists of one hidden layer and the output. The LSTM unit comprises a more complex architecture in order to determine which data will be used in each time step.

5.3 Auto-Encoders (AEs)

Automatic processing units (AEs) are neural networks that learn to compress and reconstruct data, thereby capturing its most important features. They consist of an encoder, a latent coder, and a decoder that seeks to reconstruct the input data with minimal errors. In the context of device health monitoring, AEs can be used to reduce large signal data, such as vibration signals, to improve processing capacity.

They are also used for feature extraction, which enhances fault detection performance. One variant is the denoising automatic encoder (DAE), which removes noise during reconstruction to retain only useful information. As with many aspects of our work, the usefulness of something depends on how it is implemented, but in general, AEs are great at helping improve fault diagnosis, simplifying data and performing their function while preserving important insights from that data.[6]

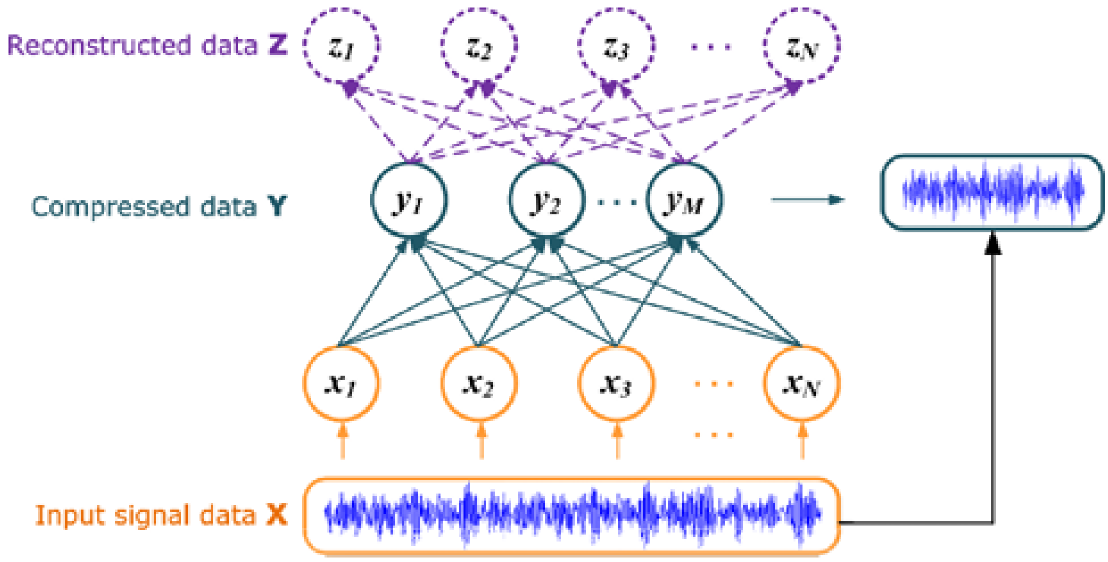

Figure 12: An Auto Encoder.

6 Deep Learning Based FD/D Methods of IMs

Deep learning(DL)has made a great stride in the area of fault detectionand diagnosis(FD/D)of induction motors(IMs), especially from sensor information in an end-to-end manner. Deep learning,unlike traditional methods of surface learning that rely on manual feature extraction techniques such as fast Fourier transform(FFT)and wavelet transform,automatically extract features. complex. hierarchies of abstract high-levelfeatures contained in multiple layers of nonlinear processing.

It helps making the process of IOT devices secure by reducing human Error and allowing large amount of data generated by IOT(Industrial IOT) NEOFIOT devicesto be process in real time. Various deep architectures, CNNs, RNNS (especiallyLSTMS). methods, such as autoencoders,and deepbelief networks, have been effectively employed to capture the complex patterns related, to the occurrence of motor failures.

The superior computational capabilities and adaptive learning ability of deep learning techniques have produced more robust,efficient andexact FD/ D systems,which becomes essential in today predictive maintenance approaches for induction motors.

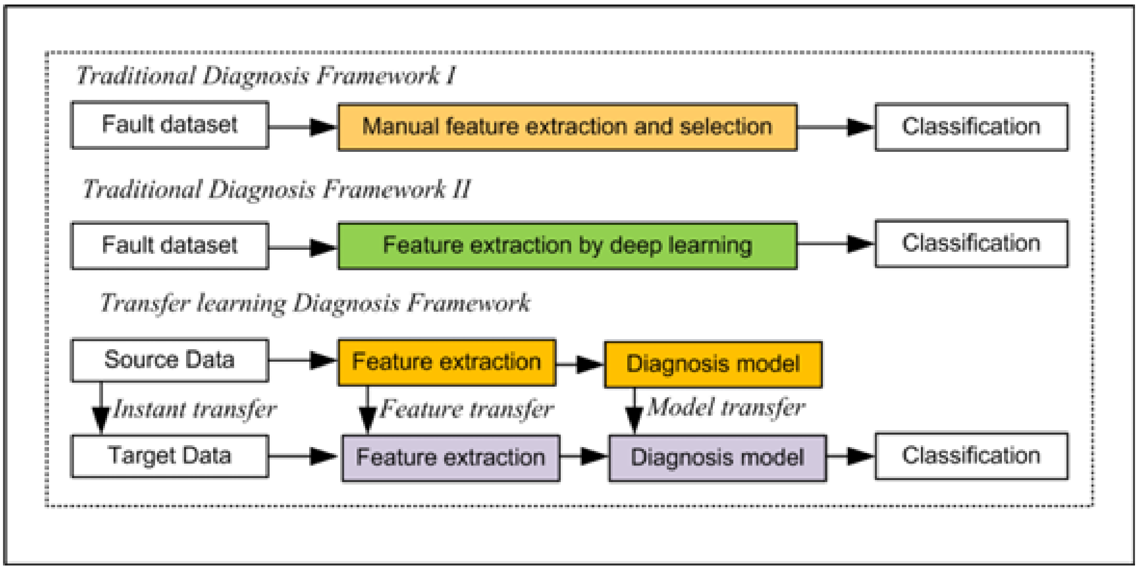

Figure 13: Diagnosis frameworks: The traditional framework (top) and the feature learning frameworks (middle, bottom)

7 Induction Motor Fault

Induction motor failures are classified into two main categories: electrical and mechanical. Electrical failures include stator winding failures, rotor bar breakage, and short circuits, all of which result in loss of electrical power and motor performance. Mechanical failures may include bearing failure, gearbox problems, misalignment, or air gap defects, all of which affect the motor's physical motion and balance. Classifying faults into categories is important for developing proper analysis and maintenance, as an electrical failure requires one method of detection/resolution, while a mechanical failure may require a different method of detection/resolution.

This allows engineers to utilize specialized monitoring and analysis, ensuring that preventative measures are taken accurately and timely, with minimal downtime and extended motor life.[8]

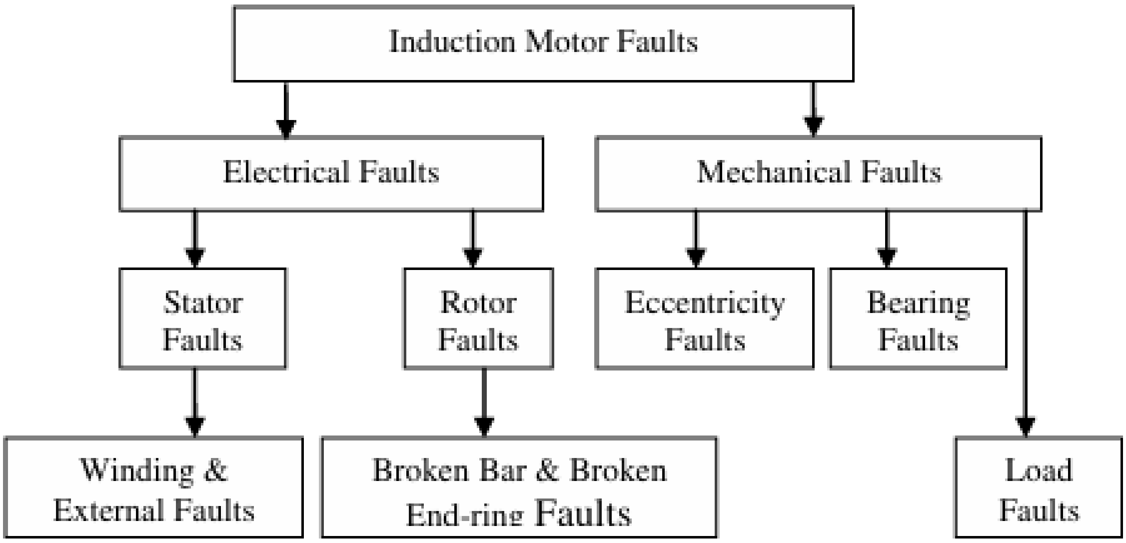

Figure 14: Classification of Induction Motor Faults.

7.1 Mechanical Faults

The mechanical faults occurrence priority is highest in the induction motor. The mechanical faults are classified as bearing fault, eccentricity fault and load fault respectively.

7.1.1 Bearing Faults

Bearing failures are among the most serious and common types of failures affecting induction motors, typically contributing to more than 40% of all failures. Bearing failures occur primarily in ball bearings or rotating elements, which provide support for the operation of motor components.

A standard bearing consists of an inner track, an outer track, and a set of rotating elements that allow movement. If defects appear in any of these components, whether pits, cracks, or fissures (collectively called defects), due to overloading, poor lubrication, contamination, or improper installation, the bearing will produce abnormal vibrations and noise.

These characteristics are unique to the bearing geometry and motor operating speed and may indicate potential bearing deterioration. Additional problems, such as rotor deflection and overheating, can also increase bearing wear and impair motor performance. Identifying and diagnosing defects in a timely manner is essential for maintenance to prevent catastrophic failures and significant motor downtime.

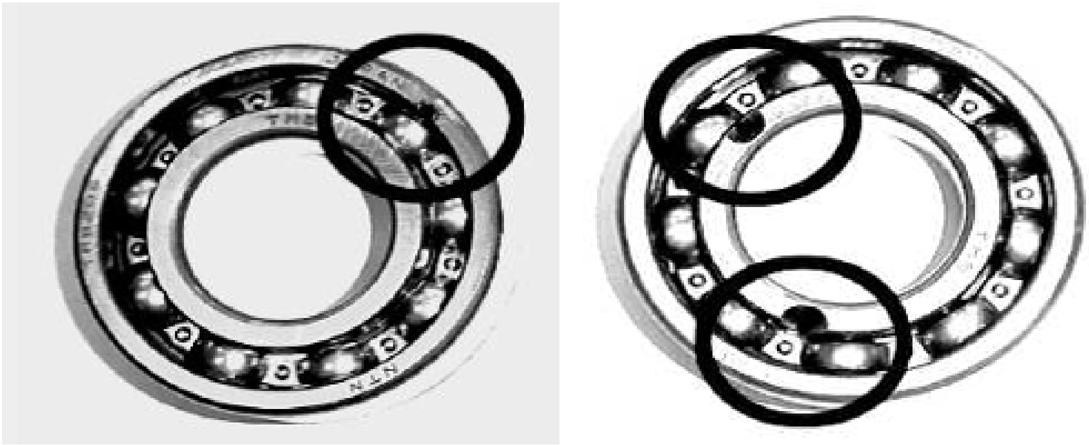

Figure 15: Artificial Bearing Defects (a) Outer Race Defect (Left) (b) Inner Race Defect(Right) .

7.1.2 Eccentricity Faults

Induction machines can exhibit eccentricity faults stemming from an unequal air gap between the stator and rotor. This can result in misalignment producing unbalanced forces, vibrations, and noise.

Eccentricity faults can be classified as either static (minimum air gap point fixed), dynamic (minimum air gap point rotates with rotor), or mixed (has both static and dynamic characteristics). Even the slightest degree of eccentricity, which is sometimes inherent due to manufacturing tolerances, can also create problems for the machine, such as rotor-stator rubbing, and can cause further damage if not handled.

7.1.3 Load Faults

Load faults in induction motors happen when an uneven load distribution or mechanical misalignment creates abnormal stresses. Typically, these faults originate from the connected mechanical components - couplings and gears.

Resulting wear can cause increased vibration, operation inefficiencies, and increased wear on the motor's components. It is imperative to identify load faults early with technologies like vibration analysis and thermal imaging to keep the reliability of the motor.

7.2 Electrical Faults

The electrical faults are classified as stator and rotor faults. The stator faults and rotor faults mainly occur in the windings.

7.2.1 Stator Faults

Stator faults happen in the windings of an induction motor, often from insulation failures between turns. Stator faults make up less than 40% of total induction machine failures. Stator faults create a shift in the balance of the magnetic field, which leads to additional heating and increased vibration.

This additional heating may lead to further insulation failures, additional heating, and possible premature failures of the bearings. It is very important to make early diagnosis of faults such as turn-to-turn, phase-to-phase or coil-to-ground faults to avoid catastrophic damage to the machine.[8]

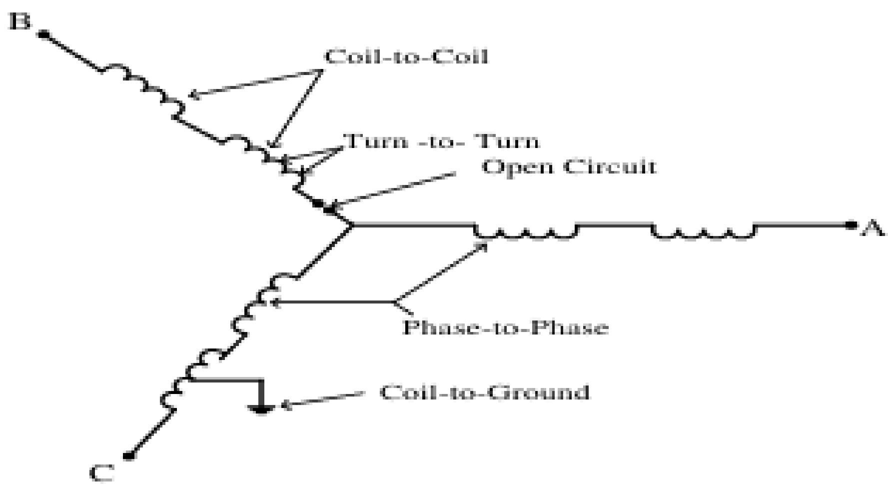

Figure 16: Graphical Representation of Stator Faults .

7.2.2 Rotor Faults

Around 10% of the total failures in induction motors come from rotor faults, which are largely associated with rotor winding defects. The main cause of rotor faults is broken rotor bars, which occur due to a pulsating load and direct on-line starting of the induction motor.

Rotor faults can cause variations in speed, torque pulsations, and, ultimately, vibrations and overheating. Other consequences include arcing from the rotor and damage to rotor laminations. A variety of stresses - thermal, magnetic, residual, dynamic, environmental, and mechanical - are generally responsible for rotor failure.

8 Fuzzy logic

Fuzzy logic is a computational method that allows consideration of uncertainty, by reproducing the way a person thinks about uncertainty in if-then rules.

It turns uncertain and noisy sensor data into information that has meaning at the time of occurrence. Instead of relying on hard decision making, fuzzy logic enables flexibility in decision making when relating to situations like predictive maintenance for induction motors. Fuzzy logic increases system reliability and efficiency because of its ability to account for different conditions, in an operating variable, relative to fuzzy logic conditions.

8.1 Fuzzy logic classification

Fuzzy logic classification is a method of fuzzy logic to classify the input data into classes based on fuzzy rules and linguistic variables. Unlike traditional binary classification approaches, which will classify data points into discrete classes, fuzzy logic classification represents uncertainty while classifying.[9]

8.1.1 Linguistic Variables

In fuzzy logic, linguistic variables act as qualitative representations of input data and output classes in natural language. They reflect the imprecision of everyday language, allowing systems to reason more similarly to humans when precise numerical values don't exist.

This allows for added flexibility in fuzzy inference systems to deal with vague or uncertain data. All in all, linguistic variables are essential in fuzzy logic to model and interpret complex real-world phenomena.

8.1.2 Fuzzy Sets and Membership Functions

Fuzzy sets are used to indicate the degree of membership of an item in a particular taxonomic category. Each linguistic variable is associated with one or more fuzzy sets, each of which has its own membership function that indicates the degree of membership of the input data point in that fuzzy set. Membership functions can take various forms, including triangular, trapezoidal, or Gaussian, depending on the nature of the data and the classification problem.

8.1.3 Fuzzy rules

Classification based on fuzzy logic uses fuzzy rules to characterize the relationship between the input variables and output classes. Fuzzy rules are stated as if-then statements; the antecedent (the then part) defines the condition and refers to the linguistic variable and fuzzy sets, and the consequent (the then part) specifies the output class. One example of a fuzzy rule is If temperature is hot and humidity is high, then classify as class.

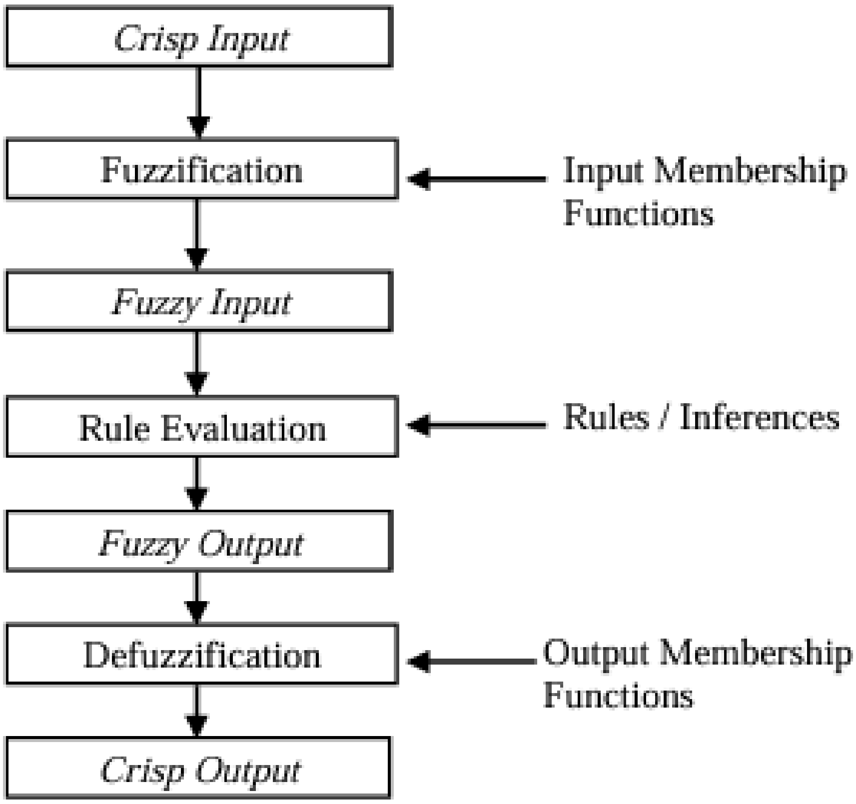

Figure 17: FIS Formulation .

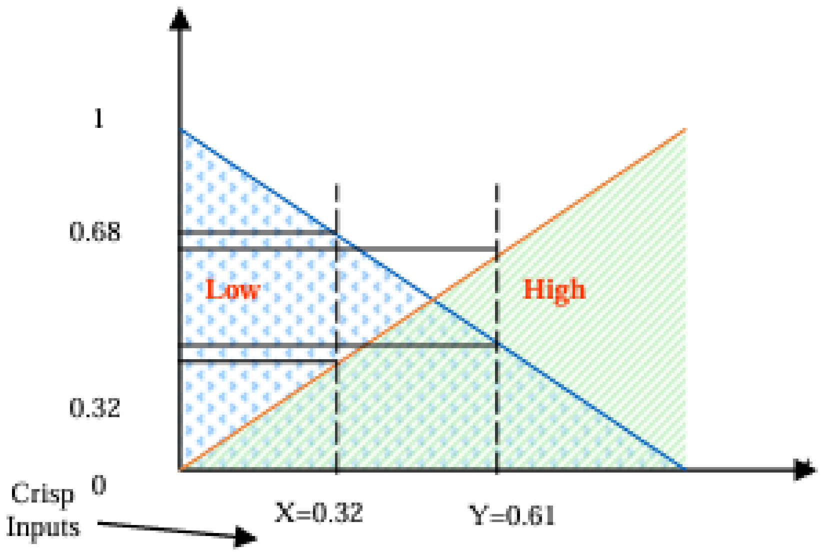

Figure 18: Fuzzy Rule Setup.[9]

8.1.4 Fuzzy Inference

Fuzzy inference applies fuzzy logic to address information uncertainty. It is based on fuzzy rules that indicate the extent to which certain values belong to output classes. Truth values are calculated based on the rule's premises, using operators such as AND, OR, and NOT, as well as aggregation methods such as max and min. The robustness of the output indicates the fuzzy belonging to the output control classes and can be used to accurately choose between multiple alternatives in indirect conditions.

8.1.5 Defuzzification

After the degrees of membership have been identified in each output class, the step of defuzzification occurs for each data point's final class assignment. Each fuzzy rule in each class has an aggregated value of the degree of membership, allowing for a crisp output value and/or class label to be assigned to the data point.

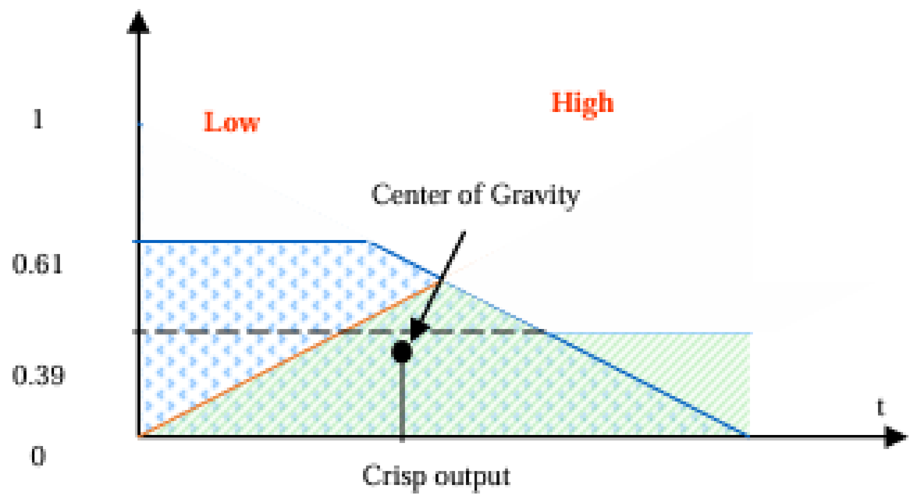

Figure 19: Fuzzy Output .

8.2 Fuzzy Logic Classification for Predictive Maintenance

Fuzzy logic classification is well suited for working with data that is imprecise and allows for interpretability through its linguistic rules, making it a great match for complex applications such as predictive maintenance, decision making, and pattern recognition. It calculates the degree of membership in fuzzy sets using functions like A(x), which evaluate inputs to fuzzy sets, and its output rules are organized as if-then statements connecting antecedent conditions to consequent outputs. Inference applies fuzzy operators (e.g. the AND operator applies the minimum function) to the degree of membership that then forms the fuzzy output.

The fuzzy output is defuzzified to give a crisp value using various methods such as the centroid method. With this system it can provide a high degree of control and accuracy of classification by mapping the sensor data (temperature, vibration, oil level) directly to maintenance actions.

Table 1: Additional linguistic variables for predictive maintenance in industrial systems

Rule	**Condition 1** (Temperature)	**Condition 2** (Vibration)	**Condition 3** (Oil Level)	**Output**
1	High	Low	Low	Maintenance is Urgent
2	Normal	High	Low	Maintenance is Urgent
3	Low	High	Normal	Maintenance is Low Priority
4	High	Normal	Low	Maintenance is Scheduled
5	High	Normal	Normal	Maintenance is Urgent
6	Normal	Low	Low	Maintenance is Scheduled
7	High	Normal	Normal	Maintenance is Urgent
8	Low	High	Low	Maintenance is Scheduled
9	Low	Low	Normal	No Maintenance Needed
...	...	...	...	...

Condition 1 (Temperature): represents linguistic variables related to the temperature of the equipment, such as High, Normal, or Low.

Condition 2 (Vibration): represents linguistic variables related to the vibration levels of the equipment, such as Low, Normal, or High.

Condition 3 (Oil Level): represents linguistic variables related to the oil level of the equipment, such as Low, Normal, or High.

Output : Represents maintenance needs based on specified conditions, including urgent maintenance, low priority maintenance, scheduled maintenance, or no maintenance needed.

8.3 Fuzzy Neural Network (FNN)

Fuzzy Neural Networks (FNNs) are hybrid systems combining the learning capabilities of neural networks with the interpretability and uncertainty of fuzzy logic. Thus, FNNs can be used effectively in predictive maintenance applications. In an FNN, the raw inputs are fuzzified into fuzzy sets in the fuzzification layer, and the neurons use activation functions in the representation of linguistic terms. For example, neurons representing the "increasing" fuzzy set would use the following sigmoidal activation function:

$$y = \frac{1}{1 + e^{-(w_1 x + w_0)}} \tag{3}$$

In this equation, w0 represents the bias and w1 denotes the weight. Instead, the complementary and Gaussian functions can be utilized for "decreasing" and "constant" fuzzy sets, respectively. This bidirectional framework enables the network to automatically determine fuzzy rules and update the associated parameters based on training data and simultaneously model the complex, non-linear relationships that occur within industry processes. The advantages of FNNs include greater robustness while dealing with uncertainties, enhanced interpretability through the use of linguistic variables, and increased accuracy when conducting fault classifications.

FNNs are important in recent predictive maintenance literature, as this technology works by converting sensor data into measures of early equipment failure, which creates opportunities for preventive action that reduce unplanned downtime, maintenance costs, and overall equipment lifecycle costs.

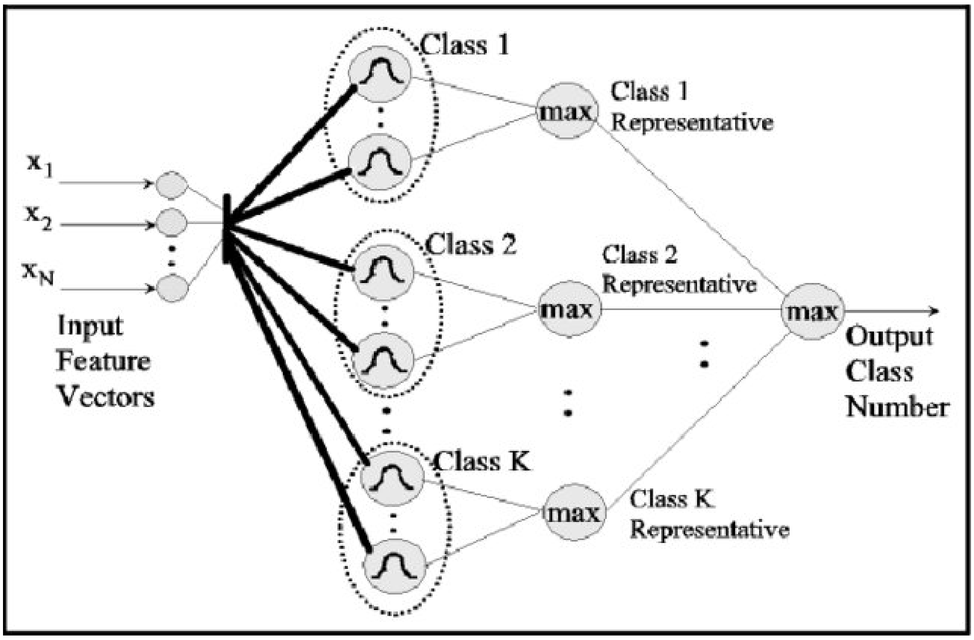

Figure 20: Fuzzy Neural Network.

9 Key technology of IoT

The Internet of Things (IoT) is the linking of physical things over networks to collect and share data. The connected things are capable of connectting with the user - whether that be a human, machine, or hybrid - to motivate, or facilitate a smarter decision-making process. The technologies that make up IoT consist of, sensors to collect data, communication protocols to exchange that data through networks, cloud platforms to store and process, smart algorithms to analyze and control processing.[7]

9.1 IoT Architecture for Intelligent Monitoring

There are five layers in the IoT (Internet of Things) architecture, which includes the Perception Layer, Network Access Layer, Network Layer, Application Support Layer, and Application Layer. The Perception Layer collects data, using sensors to do so ,then, data sends over technologies (such as Wi-Fi and Bluetooth) within the Network Access Layer; then was sends over the network, ensuring data communication solutions globally, in the Network Layer; once in the Application Support Layer, data can be processed and analyzed as needed in platforms such as cloud-computing .

the Application Layer provides user-oriented services, such as smart homes and health monitoring. Together, these conform layer capabilities to accommodate scalability and rapid development of a smart and intelligent IoT ecosystem and serve as a foundation for Cognitive IoT.

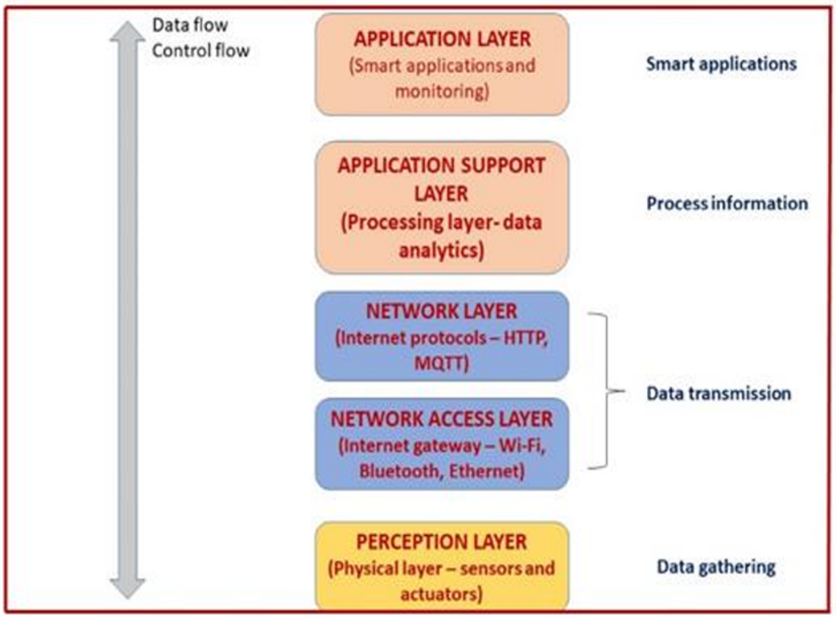

Figure 21: 5 – Layer IoT architecture .

9.2 IoT and Big Data in AI-Driven Predictive Maintenance

Predictive maintenance supported by the use of IoT and Big Data Sensors and the introduction of Big Data are revolutionizing predictive maintenance by providing real-time monitoring and sophisticated analytics. IoT sensors will provide continuous, granular sensor data from industrial equipment that greatly increases the statistical power of AI models to detect subtle changes and predict failure.

The amount of sensor data is exponentially increasing and Big Data analytics is critical in managing and analyzing the data to identify patterns, trends, and anomalies. The combination of IoT and Big Data will advance the capabilities of AI-driven models to provide timely and accurate transparent insights for maintenance to reduce downtime and improve operating efficiency.

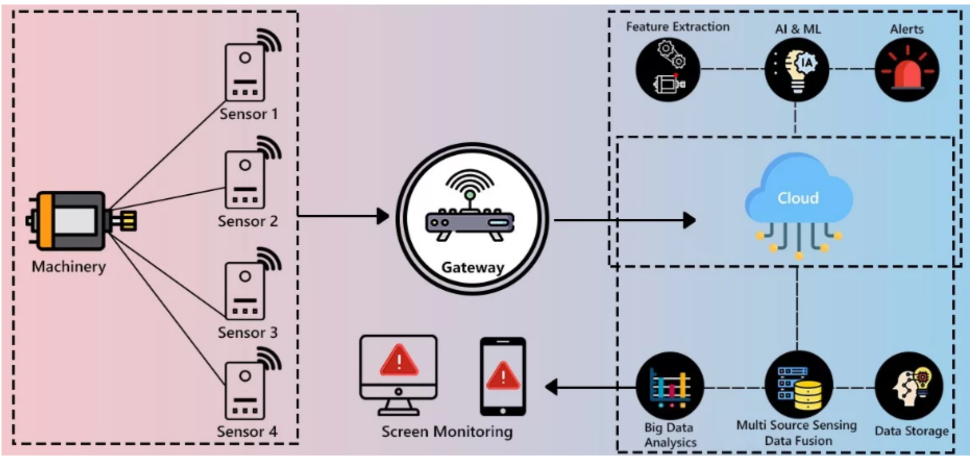

Figure 22: IoT-based predictive maintenance.

10 Conclusion

This chapter has provided a thorough examination of the basic elements that form the foundation for modern predictive maintenance practices. The chapter began by discussing Prognostics and Health Management (PHM) systems, which are important for monitoring system health and predicting failures, allowing for safety and productivity. We then transitioned into the role of artificial intelligence for predictive maintenance, explaining how machine learning or deep learning models are changing the landscape for fault detection in systems with increased complexity, parametric analysis, and a priori probability. Following, fuzzy logic classification was analyzed in the context of decision-making and explainability in uncertain situations. Lastly, IoT, a key technology, was summarized for its importance in the collection and communication of real-time data, as it applies to functions of predictive maintenance involving more advanced analytics. Together, these areas provide a strong basis for intelligent, data-driven maintenance systems in the future of industrial preparedness.

chapter 3

Material and Methods

1 Introduction

Our research methodology and techniques for fault detection in induction motors is offered in this Chapter . We described our LSTM deep learning method, which effectively models and analyses the temporal dynamics of a sensor data stream. Then we explained the diagnosis method using an Adaptive Neuro-Fuzzy Inference System (ANFIS) based on fuzzy logic for greater uncertainty and fault classification. Also important is that we introduced a combination of LSTM and ANFIS, which operates in parallel to leverage the advantages of both. The same data is processed simultaneously, and the results are then combined to produce a unified prediction with improved accuracy. and explained how the combination improves detection by combining dynamic temporal analysis with interpretable rule-based diagnosis. We have used a dataset with four labeled classes 'bearing fault', 'mechanical fault', 'normal', and 'winding fault' to detect and classify faults, and we ensure that the system we developed is capable of distinguishing between fault types and normal operation.

2 Proposed Algorithms

Our approach used an LSTM-based deep learning model to extract temporal features from consecutive sensor data, and subsequently used an ANFIS fuzzy logic model to classify the features in a way that uncertainty was handled via interpretable fuzzy rules. We provided integration through serial methods, to ensure dynamic data is translated into fault predictions for this approach. Our collaboration allows early identification of a fault and maintenance of a machine, to decrease unplanned downtime.

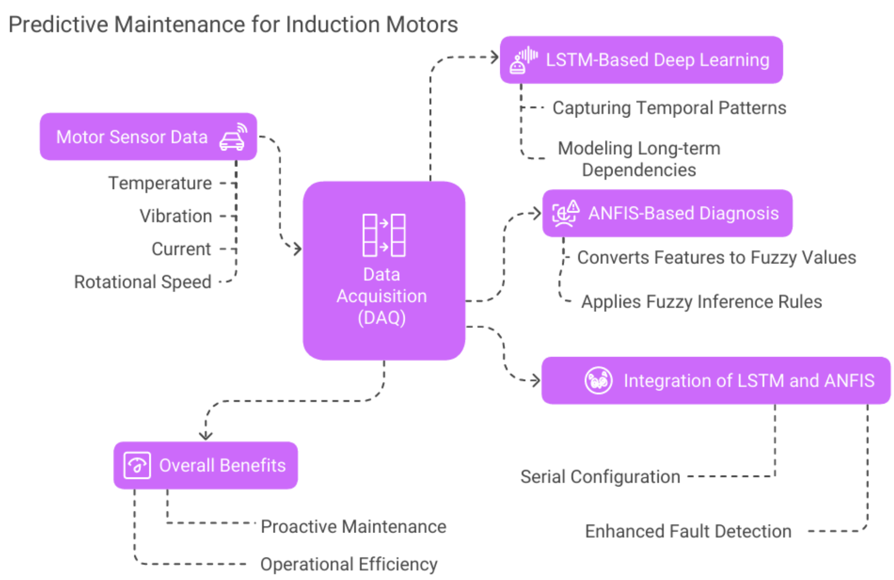

Figure 23: The Architecture of our Algorithms.

3 LSTM Approach for Fault Diagnosis

3.1 LSTM Structure

Long Short-Term Memory (LSTM) networks are a specialized kind of recurrent neural network (RNN) formulated for capturing long-term dependencies in sequential data through the use of memory cells and gating. In a stacked LSTM architecture a deep version of the standard LSTM there are multiple LSTM layers in the network. The first layer receives the raw sequence data and all subsequent layers receive the hidden state of the previous layer as input. This tiered architecture allows the LSTM network to capture increasingly abstract temporal patterns in deeper layers with greater accuracy for more challenging sequence prediction applications, including fault detection in complex systems.[10]

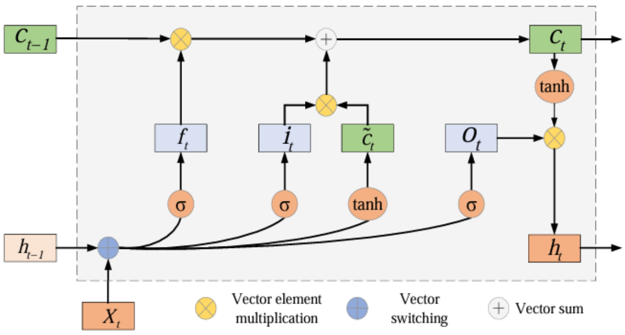

Figure 24: LSTM Network structure.

• LSTM networks use three main gates the forget, input, and output gates–to control the passage of information over time. These gates allow the network to forget non-relevant past information, add the current input values to the cell state, and only pass relevant outputs through for use in fault detection (or fault diagnosis, depending on the application).The key equations are presented as:

Forget Gate:

$$f_t = \sigma\Big(W_f[h_{t-1}, x_t] + b_f\Big). \tag{4}$$

Input Gate:

$$i_t = \sigma\Big(W_i[h_{t-1}, X_t] + b_i\Big) \tag{5}$$

$$c_t^{\text{candidate}} = \tanh\Big(W_c[h_{t-1}, X_t] + b_c\Big) \tag{6}$$

Output Gate:

$$o_t = \sigma\Big(W_o[h_{t-1}, X_t] + b_o\Big) \tag{7}$$

Cell State and Hidden State Update:

$$c_t = f_t c_{t-1} + i_t c_t^{\text{candidate}} \tag{8}$$

$$h_t = \tanh\Big(c_t\Big) o_t \tag{9}$$

Table 2: Description of LSTM Variables

Symbol / Variable	**Description**
X_t	Input vector at current time step t (e.g., sensor data)
h_{t-1}	Hidden state (output) from previous time step
w_g	Weight matrix for gate
b_g	Bias vector gate
f_t	Output of the forget gate to controls what to forget
i_t	Output of the input gate to controls what to update
o_t	Output of the output gate to controls what to output
c_{t-1}	Cell state from the previous time step
c_t	Updated cell state at time t (memory)
$\hat{c}_t$ or $c_t^{\text{candidate}}$	Candidate cell state, computed using $\tanh$ activation
h_t	Output (hidden state) at time t
σ	Sigmoid activation function (outputs values between 0 and 1)

3.2 Stacked LSTM (MLSTM)

Stacked LSTM networks are deep networks consisting of multiple LSTM layers with multiple memory cells. The first LSTM layer takes sequence data as the input, and the subsequent LSTM layer takes in the output hidden state of the previous LSTM layer, in which the architecture of stacked hidden layers deepens the model compexity while increasing accuracy. This kind of network has become a powerful tool to solve sequence prediction problems. Figure 25 shows the architecture of stacked LSTM networks without hidden layers.

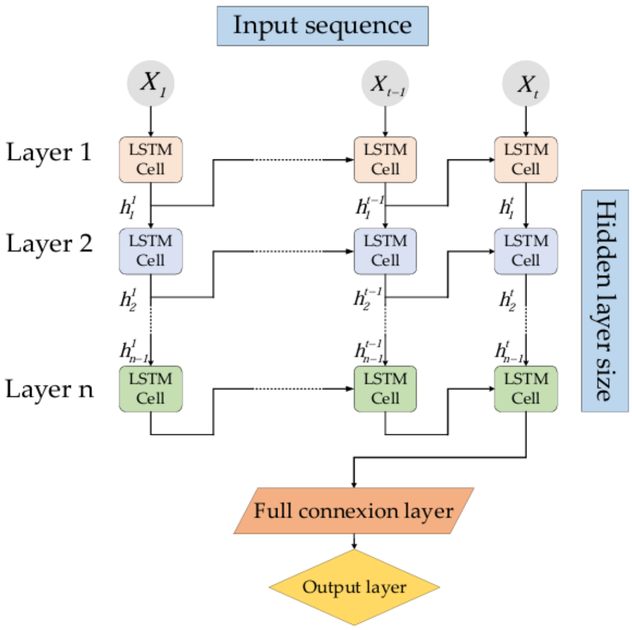

Figure 25: Stacked LSTM structure.

3.3 Bidirectional LSTM (BiLSTM)

BiLSTM is an emerging approach for addressing fault diagnosis. This technique has been used to provide more accurate results compared to other traditional methods and LSTM. In this study, a BiLSTM network is used to identify and localize multiple open-circuit faults in a three-phase two-level voltage source inverter for an induction motor drive system. Compared with LSTM, BiLSTM can obtain information from both the previous and subsequent segments. The BiLSTM architecture consists of a forward LSTM layer and a backward LSTM layer, which reverse the flow direction of the input sequence. Using LSTM twice makes the prediction results more complete and leads to increased model accuracy. Furthermore, it is worth noting that BiLSTM is a much slower approach and requires longer training times. The BiLSTM architecture is shown in Figure 26.

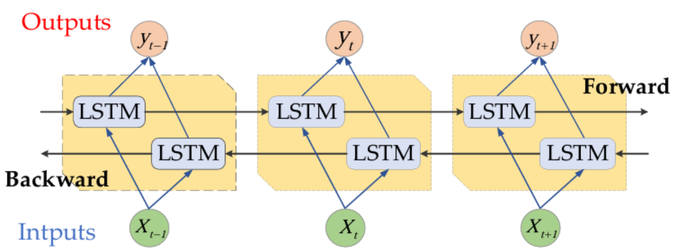

Figure 26: BiLSTM structure[10].

3.4 Evaluation Metrics

After building prediction models, several metrics can be used to evaluate the performance of models and compare them.

3.4.1 Root-Mean-Square Error (RMSE)

The root-mean-square error (RMSE) is the most commonly used performance measure for prediction tasks. The RMSE can be calculated by using (10)

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2} \tag{10}$$

where $\hat{y}_i$ and y_i are the prediction and the real output value, and n is the number of data.

3.4.2 Absolute Error (MAE)

The MAE is the other criterion used to evaluate the model performance. The MAE is expressed as (11)

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right| \tag{11}$$

3.4.3 Mean Absolute Percentage Error (MAPE)

The MAPE is one of the most common metrics used to measure the prediction accuracy of a model, and it is described as (12)

$$\text{MAPE}(\%) = \frac{1}{n}\sum_{i=1}^{n}\frac{\left|y_i - \hat{y}_i\right|}{y_i}100 \tag{12}$$

The summation ignores observations where $y_i = 0$. In general, the lower the MAPE value is, the more accurate the model is. The last metric used in this paper to evaluate the performance of the proposed algorithm is the accuracy of the prediction model, which is expressed as (13)

$$\text{Accuracy}(\%) = 100 - \text{MAPE} \tag{13}$$

3.5 LSTM-Based Diagnostic Network for Induction Motor

The topic is the design and implementation of a diagnostic and verification framework for the induction motor using a deep stacked LSTM approach. This framework combines data from an ample set of sensors, including diagnostic standard variables based on current sensors A, B, and C, along with voltage, temperature, vibration sensors, and speed sensors, to create a broader understanding of the motor's performance. In our approach, we systematically gather, normalize, and preprocess this data, and look at each sensor in its own time series or data structure to identify the most salient features.

The major component of the approach is a deep stacked LSTM, where the first layer utilizes a bidirectional approach to analyze the time-based outcomes, and the second layer continues to establish features based on the time series data. Detection of faults has been evaluated through comparison of model outputs against labeled examples of mechanical fault actions, bearing failure, coil faults, and normal behavior, using root mean squared error (RMSE) as the major cost function to direct training. Thus, through this evolutionary training structure over a defined epoch, a machine-learning algorithm will become proficient at separating faults in the sensor data, thus providing a potential approach to predictive maintenance for this example of the electric motor.

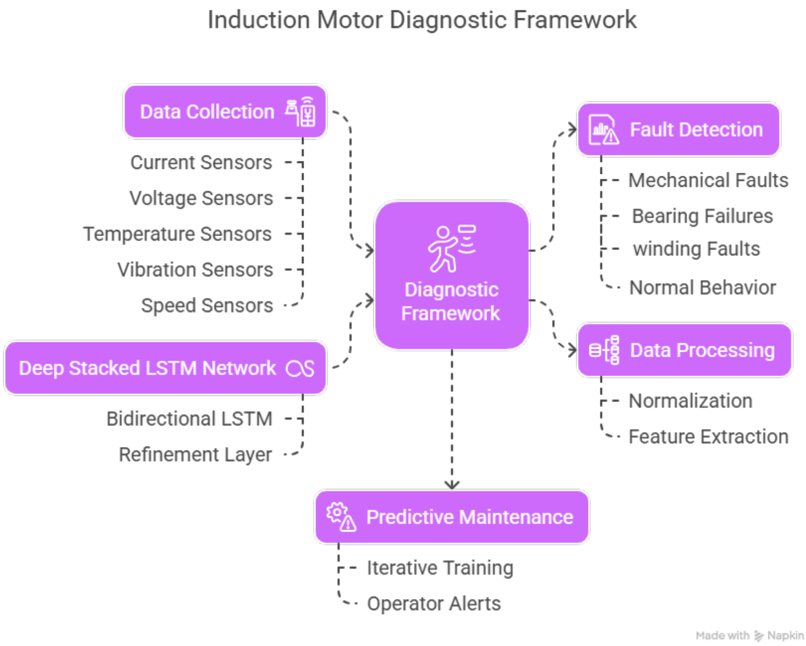

Figure 27: Structure of the fault-diagnosis module based on LSTM network.

4 Diagnosis by ANFIS

4.1 Adaptive Neural Fuzzy Inference System

The Adaptive Network-Based Fuzzy Inference System (ANFIS) is a hybrid framework that combines the interpretability and expressiveness of fuzzy inference systems (FIS) with the learning capabilities of artificial neural networks (ANN). An ANFIS takes the fuzzy rules and membership functions of a regular FIS and arranges them in large-scale to resemble a layered feed-forward network, usually five layers in total. Each layer in the ANFIS performs specific operations. The first layer is the fuzzification layer that allows the crisp inputs to become fuzzy sets. The second layer computes the degree of activation of the fuzzy rules for the input. The third layer normalizes the degrees of activation. The fourth layer computes the rule outputs according to their consequent parameters. The fifth layer aggregates the outputs of the rule to derive a final output for the system. ANFIS algorithms utilize the adaptive network structure to learn and adaptively optimize all internal variables based on input-output data, making it a great choice to describe the modeling structure of ANN and FIS for nonlinear processes.[11]

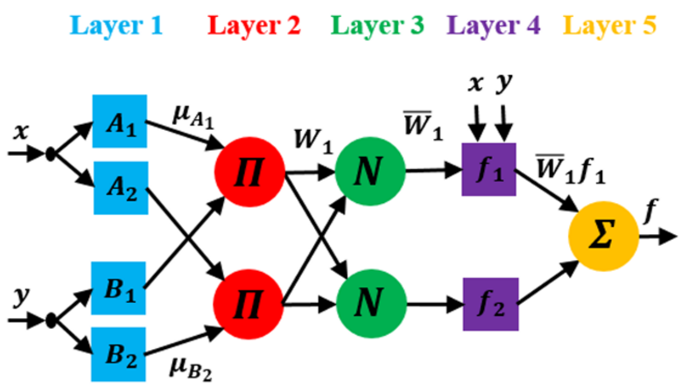

Figure 28: ANFIS architecture.[12]

• Layer 1 represents the fuzzification layer. It calculates the membership function:

$$\mu_{A_i}(x) = \frac{1}{1 + \left[\left(\frac{x - C_{A_i}}{a_{A_i}}\right)^2\right]^{b_{A_i}}} \tag{14}$$

$$\mu_{B_i}(y) = \frac{1}{1 + \left[\left(\frac{y - C_{B_i}}{a_{B_i}}\right)^2\right]^{b_{B_i}}} \tag{15}$$

where the $C_{A_i}, C_{B_i}, a_{A_i}, a_{B_i}, b_{A_i}$, and b_{B_i} are the bell function parameters.

• Layer 2 defines the rules layer. The output is the firing strength of each node:

$$W_i = \mu_{A_i}(x)\mu_{B_i}(y) \tag{16}$$

•Layer 3 denotes the normalization layer. It normalizes the calculated firing strength:

$$\overline{W_i} = \frac{W_i}{W_1 + W_2} \tag{17}$$

•Layer 4 represents the consequent layer. The output of this layer is the product of normalized firing strength and the fuzzy rules consequent polynomial:

$$\overline{W_i} f_i = \overline{W_i}(p_i x + q_i y + r_i) \tag{18}$$

where p_i, q_i, and r_i are the consequent parameter sets.

•Layer 5 defines the defuzzification layer. Its output is the overall ANFIS output:

$$f = \sum_i \overline{W_i} f_i = \frac{\sum_i W_i f_i}{\sum_i W_i} \tag{19}$$

Tuning all the adaptable parameters is the task of the learning algorithm in order to make the training data match the ANFIS output. The ANFIS model is trained to obtain the minimum Root Mean Square Error (RMSE) between the predicted values and actual measurements.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x - y)^2} \tag{20}$$

where x denotes the actual measurement, y is the predicted value and n presents the number of samples.

4.2 ANFIS-Based Predictive Maintenance for Induction Motor

When using ANFIS in a strategy for predictive maintenance, the diagnostic procedure starts with the collection of real-time sensor data from the induction motor. As RPM is a variety of measurements temperature, vibration, current, voltage, and speed no single can encompass fault-related issues. The various signals measure different aspects of motor performance temperature and vibration find mechanical or bearing issues, and measurements of current and voltage find electrical failures or coil issues; in addition, speed measures various loading conditions of the motor. The first step of using the sensor data involves using methods, such as standard scaling, to normalize the raw data to minimize variation and ensure every feature contributes to the learning process. The normalizing process is a preparation step subsequently, the normalized data into a Gaussian Fuzzification layer transfer, converting each input into fuzzy membership values using Gaussian functions.

This process discretizes continuous measurements into fuzzy membership as linguistic variables. The fuzzified membership data then contacts a series of dense layers that search the chaotic signal patterns from the various sensors and learn the various fault conditions. The model is trained on labeled data which includes many states of the motor, such as faults include mechanical, bearing, winding, and normal operation. During training, the model adaptively tunes the parameters of the fuzzy inference system. The model can purposely learn using fuzzy logic systems, and the optimization capabilities of neural networks thus, creating a powerful and effective platform for diagnostic and predictive anticipatory maintenance of induction motors.

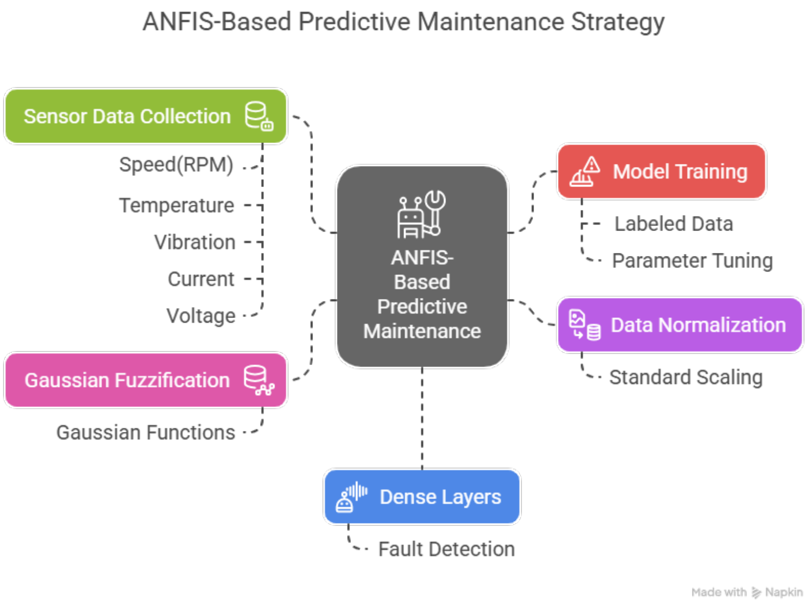

Figure 29: ANFIS-based predictive maintenance module structure.

5 Integration of ANFIS and LSTM

5.1 Rationale to build combination model ANFIS and LSTM

The justification for developing a hybrid model, using ANFIS and LSTM for the predictive maintenance of induction motors, is to take advantage of the strengths of both methods. ANFIS is highly capable in uncertain and imprecise sensor data modeling with fuzzy logic; it also allows expert knowledge to be included as linguistic knowledge rules, giving a clear understanding of interpretability.

This enables the relationship of each motor condition to be better understood, and provides insight into early deterioration or faults being present.In a more general sense LSTMs provide a foundation for modeling time-series data with time-dependencies that develop patterns or trends over much longer time periods. The hybrid ANFIS and LSTM will be capable of a robust and accurate prediction of motor performance, taking into consideration ANFIS's fuzzy learning and interpretable feature extraction, as well as LSTM's dynamic temporal modeling. This combined technique provides reliable forecasting of failures, better scheduling of maintenance by targeting potential components for maintenance, improvement of unplanned downtime, and increase in operating efficiency and lifetime of induction motor utilization[13]

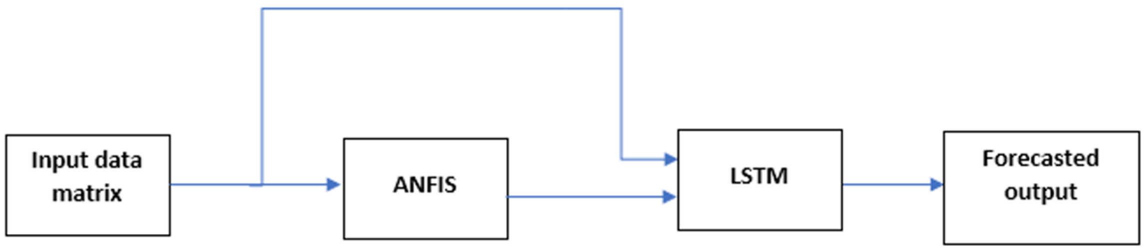

Figure 30: ANFIS-LSTM model[13].

5.2 combination Forecasting Model ANFIS-LSTM

The hybrid ANFIS-LSTM model has two stages, which will allow us to harness fuzzy logic as well as temporal learning through the two stages. An example of a stage is with the ANFIS model (traditionally 5 layers: fuzzification, rule, normalization, defuzzification, and output). The ANFIS model represents the raw sensor inputs (usually 5 parameters (current, voltage, temperature, vibration, and speed)) and maps fuzzy membership values for each input. For example, the fuzzy membership values can be associated with two linguistic sets, low and high. The ANFIS model will take the fuzzy membership values for each linguistic set and produce a set of membership degrees, such as [0.1, 0.9] for one sensor. While traditional ANFIS models will deliver only a single aggregate output that is calculated through weighted averaging of the last layer output, we will leverage the fuzzy feature representation (i.e. membership values) from intermediate layers in the hybrid model.

In the second stage, the fuzzy outputs are combined with the original sensor data to create a composite input vector for the LSTM model. In practice, the LSTM model receives six sensor inputs at each time step: five sensor features and one output (or effective fuzzy features) from the ANFIS model. The LSTM model is structured as follows: Structurally, the model consists of an input layer that accepts a six-dimensional vector, followed by multiple hidden layers—for example, a bidirectional LSTM layer (with 128 units) with fewer than 64 units per LSTM layer, in each branch—finally followed by fully connected dense layers that perform the final fault prediction. This structure enables the LSTM model to learn the temporal dependencies from fuzzy feature representation which allows for accurate and robust prediction for predictive maintenance of induction motors.

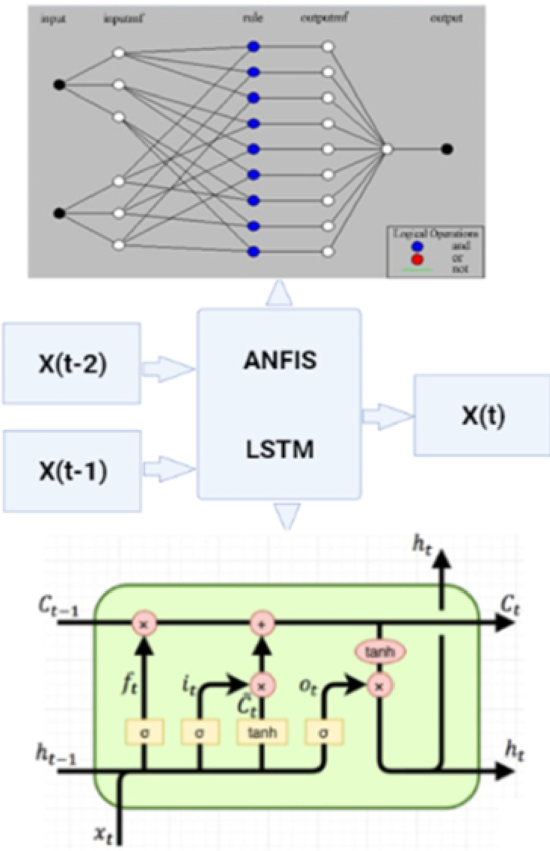

Figure 31: Flowchart of the ANFIS and LSTM model [14].

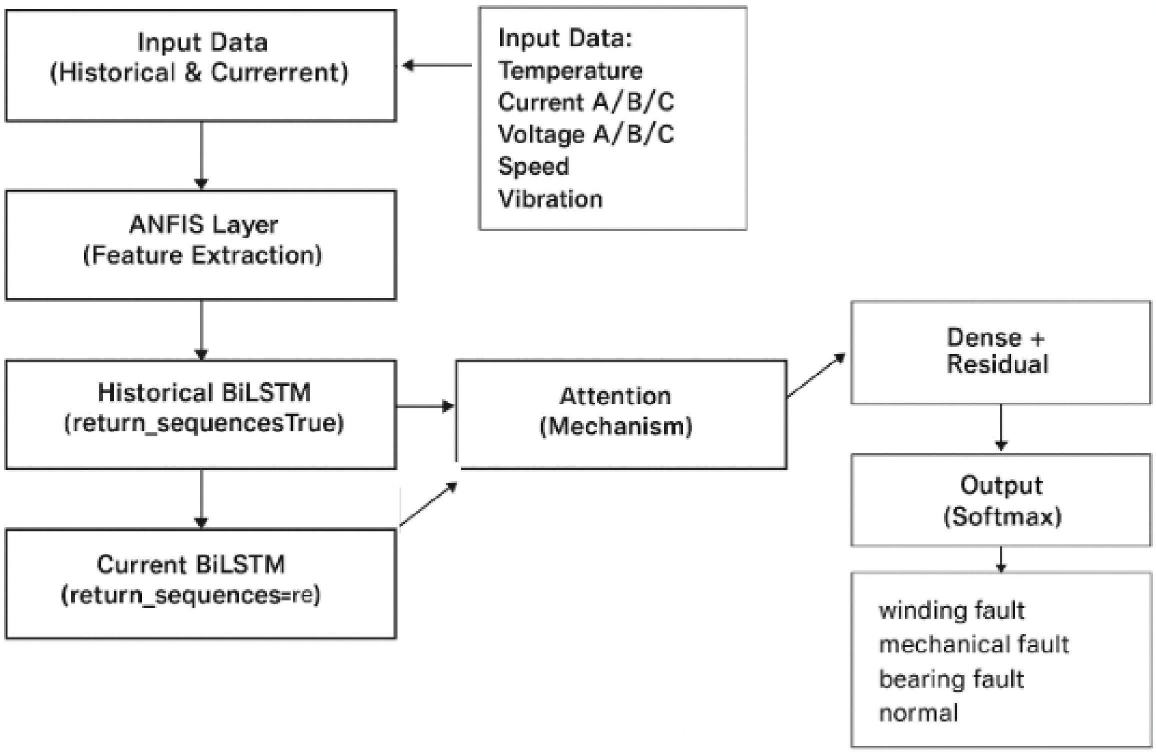

Figure 32: Flowchart of the ANFIS and BiLSTM model

6 Methods and Techniques for Data Collection and Analysis

Successful predictive maintenance is based on gathering high-frequency real-time data via the Internet of Things (IoT) sensors, historic maintenance records, and operating logs. These methods record both operational activity and context data, both of which are important for the advanced detection of anomalies and prediction of failures.

This data is then analyzed using sophisticated statistical techniques and machine learning models to generate insights that can be used to make better maintenance scheduling decisions to maximize maintenance efforts and minimize downtime.

6.1 Data Collection Techniques

Predictive maintenance data is gathered from a hybrid of IoT sensors, maintenance history data, and operational logs. IoT sensors are placed on critical machinery and track machine parameters like temperature, vibration, pressure, and rotational speed, in real time.

This data is gathered with high fidelity giving a close-to-reality picture of the equipment performance and allowing early detection of faults, as well as optimal timing of maintenance activities.

1- Sensor Data Acquisition: Wireless sensor networks enable rapid, high-frequency data collection using devices like accelerometers to monitor vibrations which helps detect wear and potential failures in rotating equipment.

2- Historical Maintenance Data: Historical maintenance records provide vital training data for ML models to uncover failure patterns and evaluate maintenance effectiveness.

3- Environmental and Operational Data: Environmental and operational data, like temperature and load, are crucial for contextualizing machinery performance and enhancing predictive model accuracy.

7 Intelligent fault detection framework based on IoT, cloud AI

Structured as a multi-layer architecture, the intelligent fault detection framework synthesizes the Internet of things (IoT), cloud computing and Artificial Intelligence (AI) to monitor the operation of machines in real time and make smart decisions based on data. Sensors are used for data acquisition, communication technologies enable seamless communication, and models organize data to deliver meaningful analyses.

It provides interactive visualizations and statistical reporting to facilitate predictive maintenance and operational decision-making.

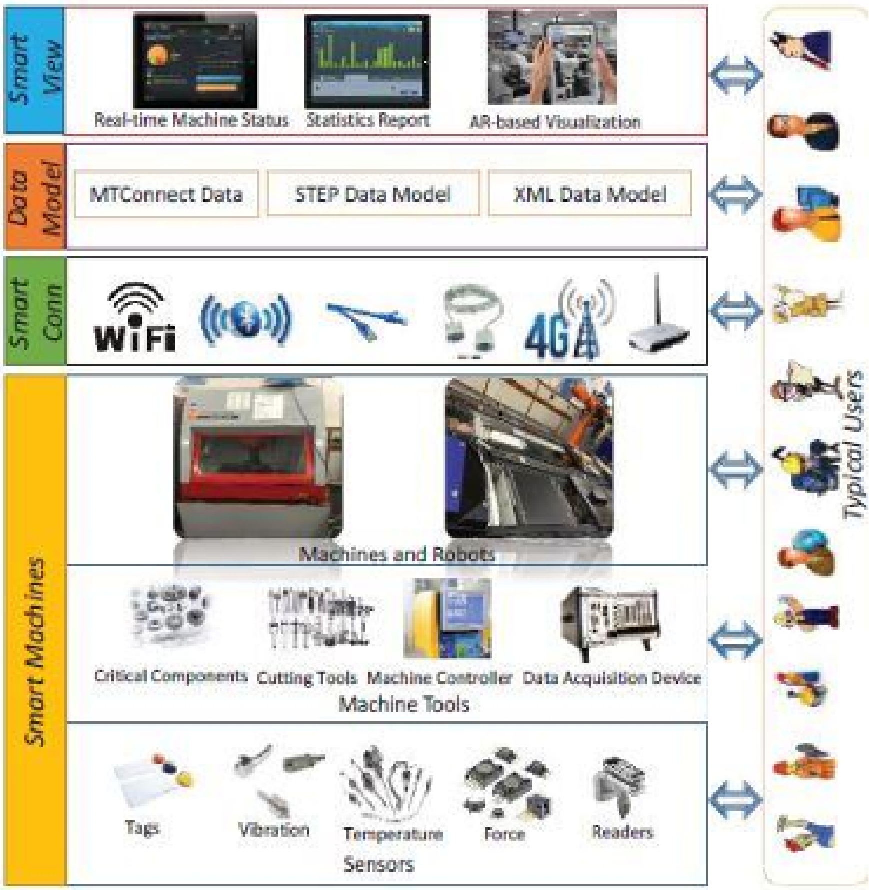

Figure 33: Architecture of IoT-enabled monitoring system[7].

8 Blynk's(IOT) Role in Translating Hardware Data into Software Insights

Blynk is an important link in the process of predictive maintenance for three-phase motors. In this role, Blynk serves as the connection between the ESP8266 microcontroller and the analytics systems running on a Python environment. The ESP8266 gathers real-time data using various sensors that measure current, voltage, vibration, and speed. As it collects this data, it sends this data to the Blynk Cloud through virtual pins. Virtual pins are software-based channels that allow the software to abstract the hardware aspects of the channels, which is essential in the ability to manage data . The Blynk library and the ESP8266 communicate over a secure TCP connection. Once the data is received at the Blynk Cloud, the Python applications can retrieve them using Blynk's RESTful HTTP application programming interface (API). The Python scripts can send HTTP GET requests to specific endpoints where the latest sensor readings are placed. This allows the predictive maintenance analytics programming to handle the decision-making and analytics processes in real-time .

Blynk also supports two-way communication, presenting developers the capability of sending alerts and/or commands back to the ESP8266 from the Python applications via the Blynk platform. This two-way communication, especially when structure to send alerts to the users would allow them to address anomalies straight away or to take immediate corrective actions at hand. Blynk's cloud infrastructure temporarily stores the incoming data, simply for the fact that their is a need for periodic periodic review of the data . For some systems this data storage will allow access for a period, even if the data communication is stopped from the ESP8266. [15]

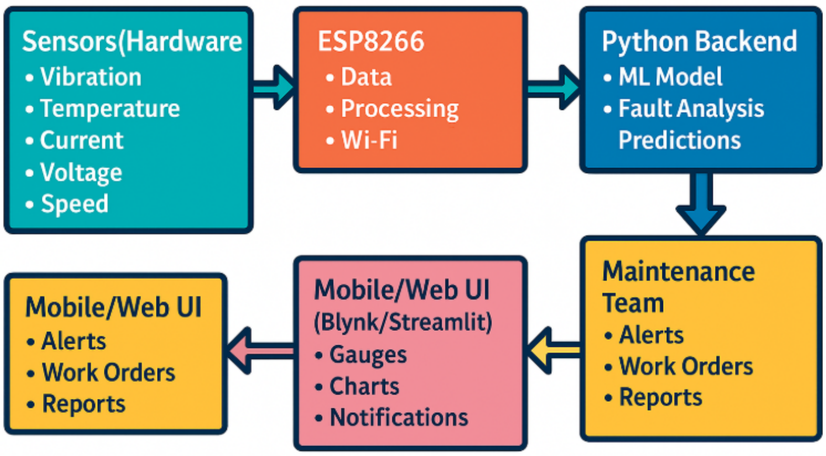

Figure 34: diagram of the predictive maintenance system.

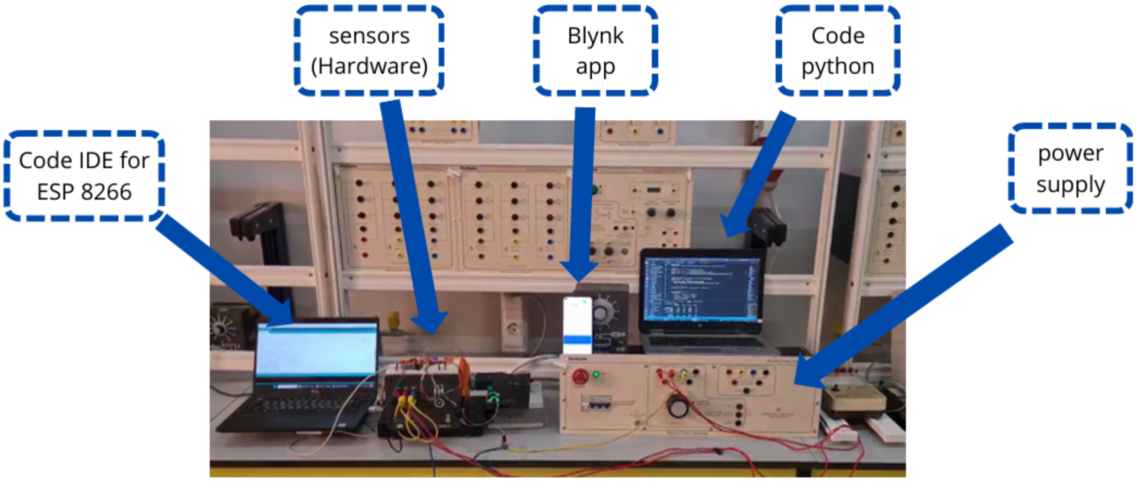

Figure 35: Architecture Diagram of the IoT-Based Predictive Maintenance System

9 Conclusion

In this chapter, we have formed a hybrid forecast model for predictive maintenance of induction motors that utilizes the interpretability of ANFIS and the temporal modeling ability of an LSTM model. Use of ANFIS in our first stage utilized the raw sensor data of induction motors obtained from current, voltage, temperature, vibration, and speed, and converted it to fuzzy membership values, losing some uncertainty but also harnessing expert knowledge via linguistic rules. Typical ANFIS models output one crisp value, however, we kept the membership values which then added to our features.The fuzzy values (outputs from the ANFIS) were conjoined with the original sensors to make our joint input. As six-dimensional input (five derived from sensors and one that came from ANFIS), the LSTM now had increased potential to learn time dependent features via layers. The LSTM contained an input layer, two hidden layers consisting of one LSTM bidirectional and one LSTM standard layer, and dense layers giving the fault predictions.We have established that a prediction model comprised of fuzzy logic–based inference from ANFIS and sequential learning via LSTM can possible strengthen our predictive maintenance practice in terms of overall possible operations and decreased downtime.

chapter 4

Experimental Setup and Results

1 Introduction

Chapter 4 presents a comprehensive experimental analysis of our three predictive maintenance models, LSTM and ANFIS, and their integration into a sequential scheme. First, we evaluate the performance of a single LSTM network on our four-class dataset (bearing failure, mechanical failure, normal failure, and winding failure), measuring its ability to capture temporal failure patterns. Accuracy, precision, recall, and F1 score are reported to measure its ability to capture temporal failure patterns. Next, we measure the results of the ANFIS diagnostic model in terms of performance and interpretable rule-based classifications. We also evaluate the integrated ANFIS-LSTM model, demonstrating how sequential feature extraction using LSTMs and fuzzy inference using ANFIS contribute to effective failure detection. For each method, we present confusion matrices and results for each class to illustrate their strengths and weaknesses. Finally, we compare the three methods and show that the hybrid model achieves the best balance between sensitivity and specificity. In this analysis, we validate the effectiveness of the proposed algorithms for predicting induction motor faults in real time, by claiming this performance.

2 Dataset

For our experiments, we constructed a time-series dataset where each record corresponds to a single timestamped snapshot of motor operation. We collected twelve features per sample:

T	I_a	I_b	I_c	V_a	V_b	V_c	Temp	Vib	Speed	RUL	Type
25-04	10.20	9.93	9.78	372.83	384.94	362.00	41.56	1.99	1719.94	578.39	normal
25-04	10.06	10.30	9.65	388.92	387.97	367.36	41.09	2.20	1708.98	528.86	normal
25-04	12.17	12.97	12.79	346.80	400.46	342.52	39.02	6.00	1487.50	276.53	mechanical fault
25-04	12.77	11.84	13.85	391.37	405.94	379.12	40.36	6.00	1487.50	135.21	mechanical fault
25-04	9.85	14.62	16.69	363.92	376.30	361.85	65.00	2.33	1736.22	87.75	winding fault
25-04	10.97	12.10	15.20	387.53	380.40	366.37	55.00	5.00	1671.35	12.62	bearing fault
25-04	13.23	13.33	12.79	401.50	373.16	366.69	55.00	5.00	1753.56	256.68	bearing fault
25-04	10.43	14.27	16.68	350.99	375.86	343.65	65.00	2.13	1690.67	45.99	winding fault

Table 3: Operating and fault data

• Our dataset includes:

variable	Description
time stamp	Precise date and time of the reading.
current a, current b, current c	Phase currents (A) measured on the three motor windings.
voltage a, voltage b, voltage c	Phase voltages (V) on each winding.
temperature	Bearing or stator temperature (°C).
vibration	Overall vibration intensity.
speed	Rotational speed (rpm).
remaining useful life	RUL estimate (hours) from our degradation model.
fault type	Categorical label specifying one of four classes: “normal,” “bearing fault,” “mechanical fault,” or “winding fault.”

Table 4: Defining the variables used in the dataset

3 Experimental settings

3.1 Libraries

In our implementation, the Python language is used relying on a set of libraries that includes:

- TensorFlow
- Keras
- Pandas
- SciPy
- Numpy
- time , json , serial, datetime , os , warnings
- stats , signal , scipy.stats.zscore
- scikit-learn
- Matplotlib , seaborn

We decided to use Python as our main programming language to meet our needs for programming. Python is a good choice for jobs involving machine learning and deep learning because to its vast ecosystem of tools and frameworks.

This code sets up the work environment for a three-phase motor fault detection project using deep learning. It imports:

- Python standard libraries (Time, json, serial, datetime, os, warnings) which will be used for core operations and Input/Output.
- NumPy and pandas for the fast numerical calculations and structured data storage.
- SciPy used for signal processing and statistics regarding the sensor readings.
- Matplotlib (Agg backend) and Seaborn for the plotting in headless mode and data visualizations.
- TensorFlow,Keras, for building and training deep learning models (LSTM, Attention layers, ANFIS integration).
- scikit-learn, for the data preprocessing (scaling, one hot encoding), splitting, metrics and evaluation.
- tkinter, optional as a standalone unit testing interface.

All of the above libraries will allow the effective ingestion, preprocessing, and development of fault prediction models for the IoT data stream, and performance reporting for predictive maintenance of induction motors.

3.2 Hardware

Table 5: Hardwar used.

Sensors	**Microcontroller Interface**
• Vibration Sensor (MPU6050 or industrial piezo-electric accelerometer) • Temperature Sensor (Thermocouple, RTD, LM35, DS18B20) • Current Sensor (ACS712 or current transformer) • Speed Sensor (Hall-effect, optical encoder, tachometer)	ESP32 : • ADC pins (e.g., GPIO34, GPIO35) for analog sensors (vibration, temperature, current) • Digital pins for one-wire or pulse sensors (e.g., DS18B20, hall-effect speed sensor) • I^2C: SDA = GPIO21, SCL = GPIO22

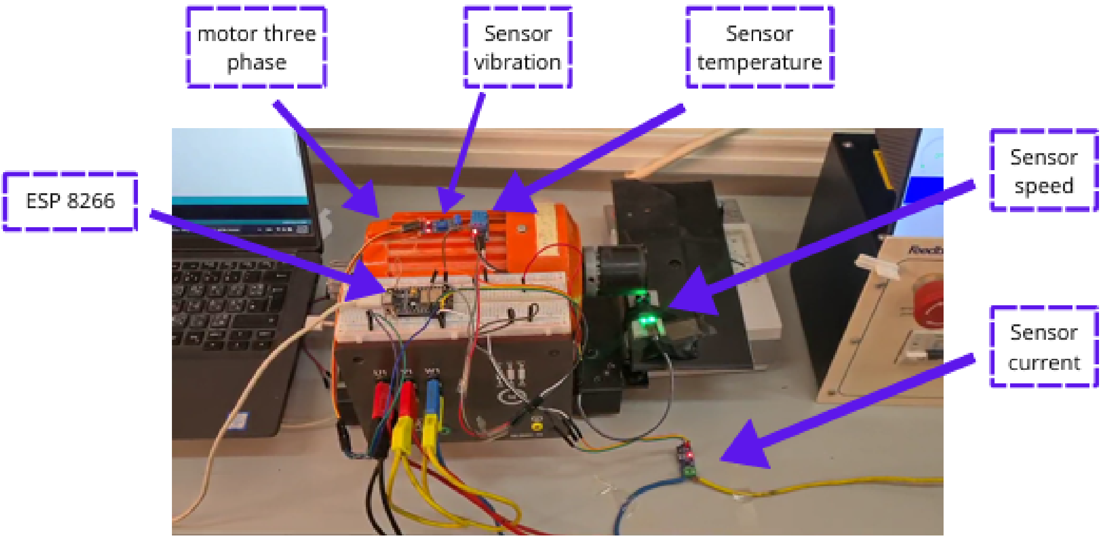

Figure 36: Electronic circuit design.

4 The proposed method and Results

4.1 Experimental protocol

We divided the dataset into 80% for training and 20% for testing, and We used it to the models we mentioned in Chapter 3. Now, we will examine the results of each method and the adopted architecture.

4.2 Performance metrics

To evaluate the performance of a model, several metrics are considered:

✦ Accuracy

- Definition: Accuracy is the proportion of true results (both true positives and true negatives) among the total number of cases examined.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

- Importance: Provides a general measure of how often the model is correct but can be misleading if the dataset is imbalanced.

✦ Recall

- Definition: Also known as sensitivity or true positive rate, measures the proportion of actual positives that are correctly identified by the model.

$$\text{Recall} = \frac{TP}{TP + FN}$$

- Importance: High recall is important in applications where missing a positive case (a fault) is very costly.

✦ Precision

- Definition: Precision measures the proportion of positive identifications that are actually correct.

$$\text{Precision} = \frac{TP}{TP + FP}$$

- Importance: High precision is important in applications where the cost of a false positive (false alarm) is high.

✦ F_1 Score

- Definition: The harmonic mean of precision and recall, providing a balance between the two metrics.

$$F_1 Score = 2\frac{\text{PrecisionRecall}}{\text{Precision} + \text{Recall}}$$

- Importance: Useful when we need to balance precision and recall, particularly in situations with uneven class distribution.

Where:

- **TP (True Positives)**: correctly predicted fault samples.
- **TN (True Negatives)**: correctly predicted normal samples.
- **FP (False Positives)**: normal samples mistakenly flagged as faults.
- **FN (False Negatives)**: fault samples missed by the model.

5 The Results

In this section, we give the specific results obtained for each model, shown via tables, matrices and curves. We show an overall summary table with the quantitative performance metrics (accuracy, precision, recall and F1-score), where we summarize each model's performance in the test set. To allow further interpretation and analysis we include confusion matrices for all four fault categories, as these matrices represent all true positives, true negatives, false positives and false negatives. Overall, this will give a good idea of the overall behaviour of the models, including areas for refinements.

6 Performance of LSTM

After applying Long Short Term Memory (LSTM) training for a total of 20 epoch, we achieved the following results, which are shown below:

Table 6: Regression Performance Metrics on the LSTM

Metric	Value
Test Mean Absolute Error	0.22
Test Root Mean Squared Error	0.47
Test R^2 Score	0.83
Test Regression Accuracy	0.83

The results of the Confusion Matrix for LSTM are shown in Table 7. The Confusion Matrix shows precision, recall, and F1 score. Precision tells how accurate the positive predictions are, recall tells how many true positive instances were found, and the F1 score tells us a balanced assessment metric that combines the precision and recall measures.

Table 7: Classification Report of LSTM

Class	Precision	Recall	F1-score	Support
bearing fault	1.00	0.67	0.80	3
mechanical fault	0.67	1.00	0.80	2
normal	0.67	1.00	0.80	2
winding fault	1.00	0.50	0.67	2
accuracy			0.78	9
macro avg	0.83	0.79	0.77	9
weighted avg	0.85	0.78	0.77	9

7 Performance of ANFIS

The results obtained using the ANFIS model will be presented in the following format:

Table 8: Regression Performance Metrics on the ANFIS

Metric	Value
Test Mean Absolute Error	0.39
Test Root Mean Squared Error	0.99
Test R^2 Score	0.04
Test Regression Accuracy	0.04

With respect to our model on the test set, its analyses are an average of about 0.39 units away from true values (MAE) and has penalties for its rare larger errors (RMSE ≈ 0.99). However, with an R^2 of just 0.04 which means it explains only 4% of the outcome's variability and gruesomely low regression accuracy it performed only marginally better than always predicting the mean. It is a relatively weak predictor, but this simple baseline is valuable: it provides simple and interpretable metrics in the original units of measures and provides a transparent benchmark on which you can evaluate the benefits of adding features, or trying more flexible models such as those that accommodate the data structure.

The outcomes utilizing the ANFIS model will be addressed in table 9 with the accuracy value highlighting the whole of the model. The F1 score will be provided per class to bring together precision and recall, creating an average measure.

Table 9: Classification Report of ANFIS

Class	Precision	Recall	F1-score	Support
bearing fault	0.77	0.70	0.73	47
mechanical fault	1.00	1.00	1.00	60
normal	0.81	0.93	0.87	61
winding fault	0.56	0.47	0.51	32
accuracy			0.82	200
macro avg	0.78	0.78	0.78	200
weighted avg	0.82	0.82	0.82	200

8 Performance integration of ANFIS and LSTM

Upon implementing the integration of ANFIS and LSTM model, we achieved highly favourable and satisfactory results.

Table 10: Regression Performance Metrics on the integration of ANFIS and LSTM

Metric	Value
Test Mean Absolute Error	0.03
Test Root Mean Squared Error	0.24
Test R^2 Score	0.95
Test Regression Accuracy	0.95

By combining the ANFIS with the LSTM component, our hybrid approach obtained excellent classification performance: the mean absolute error of only 0.03 class, and an RMSE of 0.24 class indicates nearly all predicted classes are exactly correct or within a quarter class of their true label. An R^2 of 0.95 indicated that the model explained 95% of the variance in the actual classes, and a similar regression style accuracy of 0.95 corresponds to a roughly 95% correct classification rate.

Table 11: Classification Report the integration of ANFIS and LSTM

Class	Precision	Recall	F1-score	Support
bearing fault	1.00	0.97	0.99	38
mechanical fault	1.00	1.00	1.00	50
normal	1.00	0.97	0.98	62
winding fault	0.94	1.00	0.97	44
accuracy			0.98	194
macro avg	0.98	0.99	0.98	194
weighted avg	0.99	0.98	0.98	194

The ANFIS+LSTM model reliably demonstrates great fault detection: it never mislabels mechanical failures, it identifies both normal and bearing fault conditions almost perfectly, and it catches even the most complicated cases of winding faults with only very few false alarms. This consistent, high-fidelity performance allows maintenance teams to trust alerts, only schedule interventions when interventions are truly needed, and avoid both costly unplanned interruptions and unnecessary inspections. Practically, this process results in smoother operations, longer equipment lifetimes, and significantly less maintenance overhead; which is the hallmark of any effective predictive maintenance system.

9 Confusion matrix

We now report and compare the confusion matrices for each experimental case. A confusion matrix details each model's predictions vs true labels and the number of correctly predicted instances; true and false positives and true and false negatives. The confusion matrices enable further detail of the model performance in terms of which fault types can be most confidently detected, in which situations misclassification occur, and the case wise metrics of all classes.

This evaluation of our models is beneficial since it confirms accuracy while demonstrating strengths and weaknesses; therefore, adjustments can be made to rectify weaknesses and increase the robustness of our predictive maintenance solution.

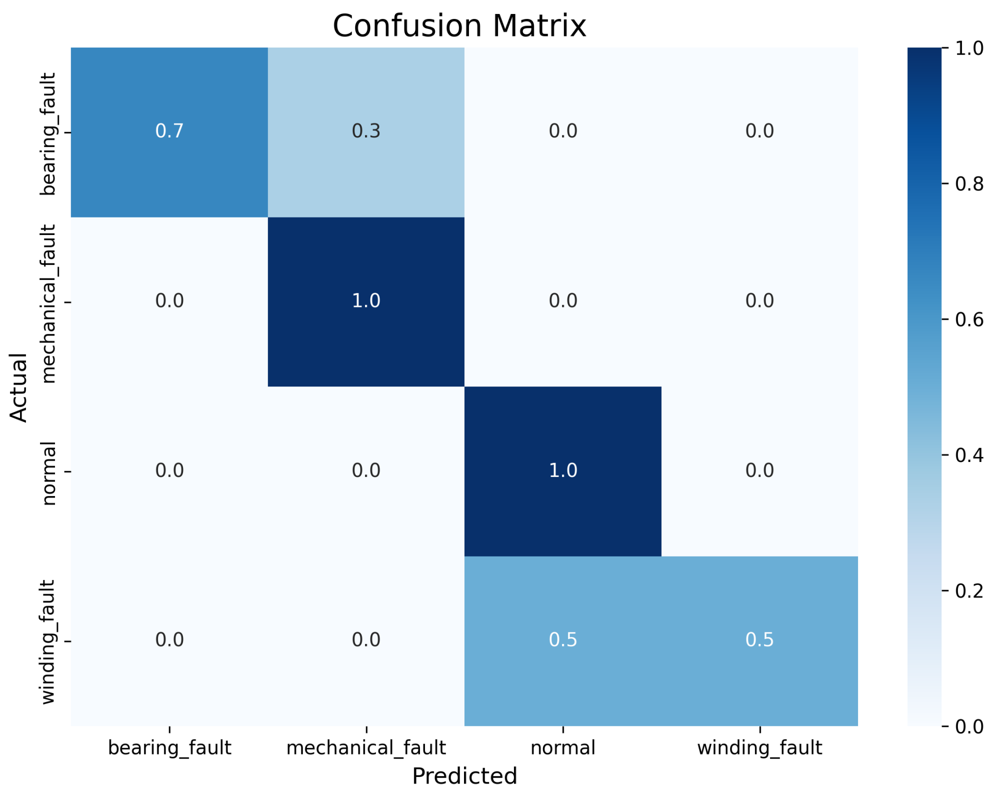

Figure 37: Confusion Matrix of LSTM.

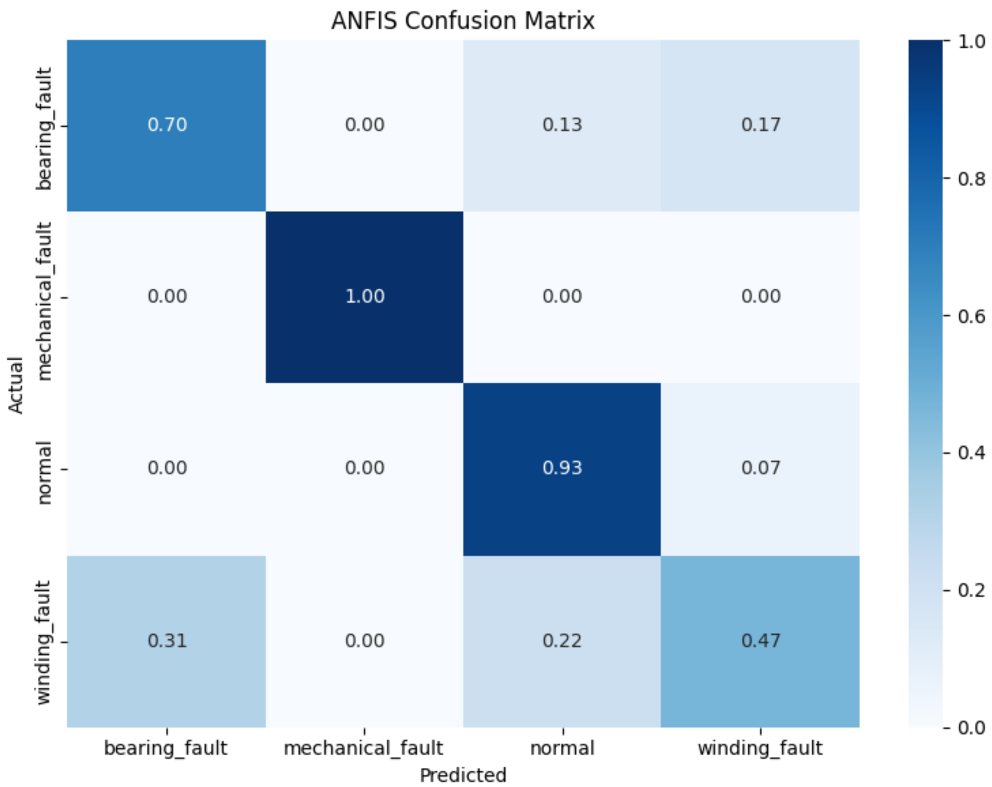

Figure 38: Confusion Matrix of ANFIS.

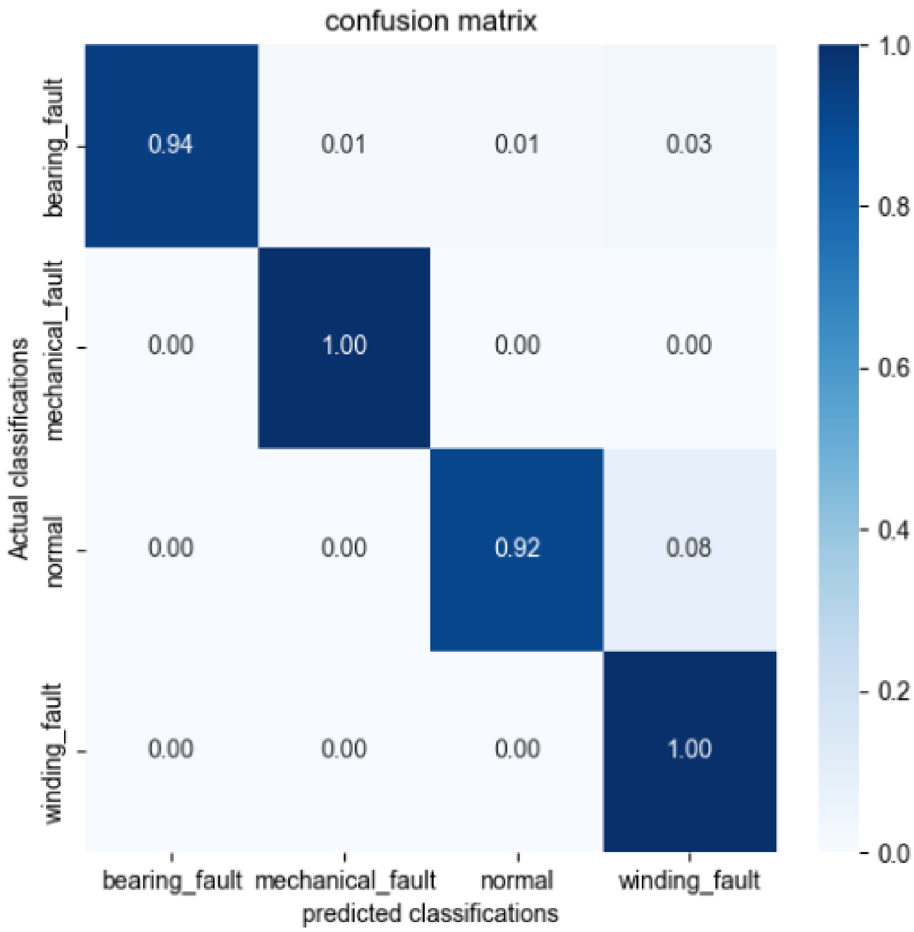

Figure 39: Confusion Matrix of integration of ANFIS and LSTM.

•The confusion matrix for" integration of ANFIS and LSTM " shows that the model correctly designated nearly all of the fault types in each class; it got perfect scores for mechanical and winding faults, 94 % for bearing faults(with only minor misclassifications), and 92% for "normal" (with 8% remaining confused with winding faults). Given the performance in each class, the classifier can be trusted to detect critical failures with few false positives. Any remaining errors tightly delineate where additional data or feature engineering could eliminate the last errors.

10 The Curves

To perform predictive maintenance on the induction motor, presented loss vs epoch curves for the training and validation sets separately. In each figure, we see training loss dropping as the model captures underlying structure when learning, while validation loss similarly decreases before flattening (or slightly increasing); this means we have not had severe overfitting. By comparing curves across models we can assess which model converges fastest, which maintains the smallest gap between training loss and validation loss, and which may be on the point of over fitting. In total, these visualizations offer learning dynamics in a clear epoch by epoch fashion and help inform our decisions for early stopping, hyper parameter tuning, and architectural change to optimize model performance on previously unseen motor fault data.

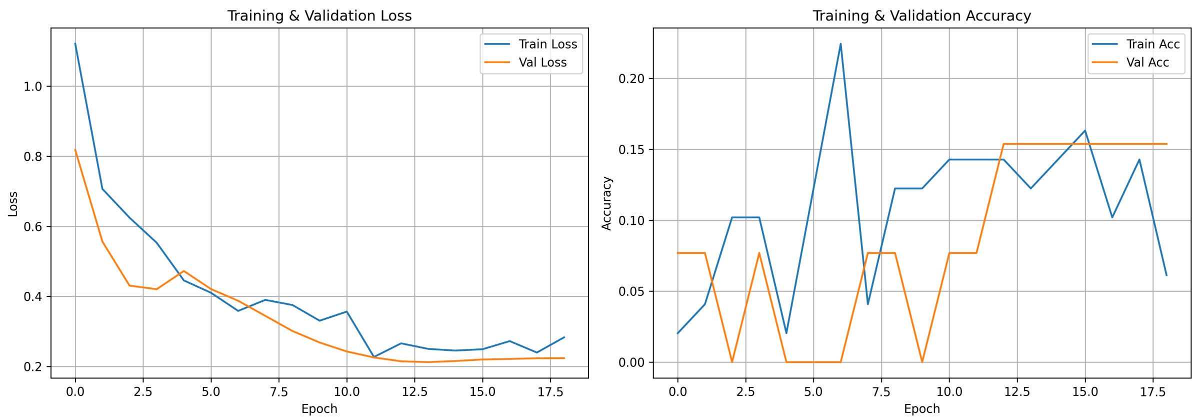

Figure 40: Curve of loss and accuracy of LSTM.

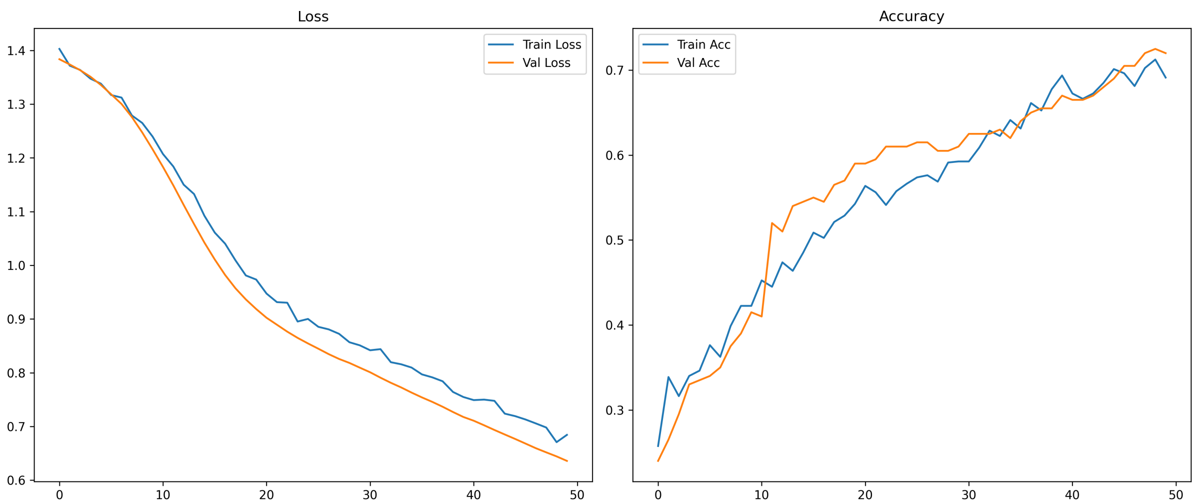

Figure 41: Curve of loss and accuracy of ANFIS.

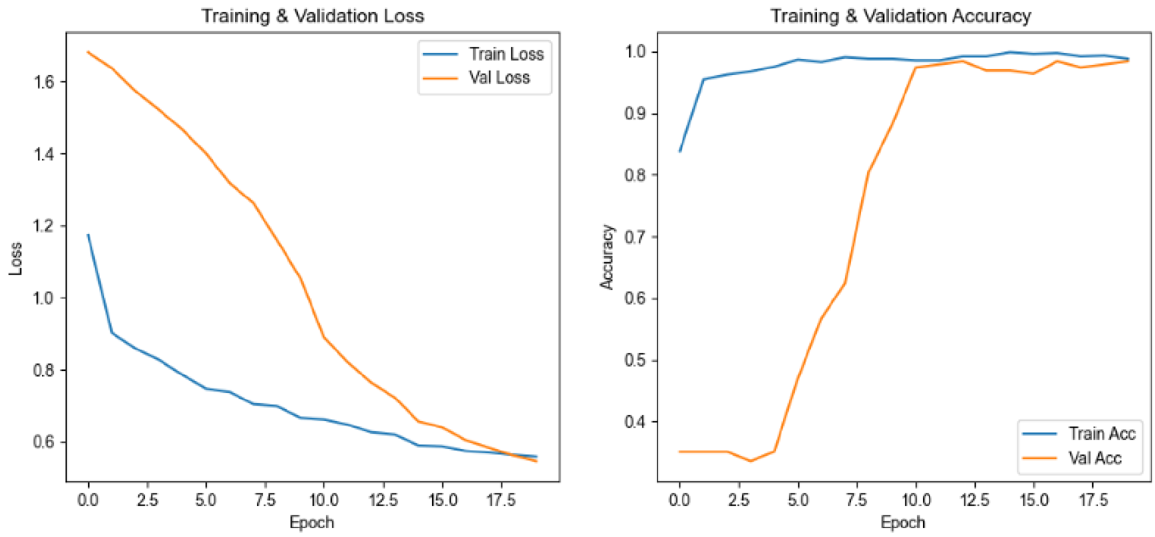

Figure 42: Curve of loss and accuracy the integration of ANFIS and LSTM.

11 Blynk IoT mobile

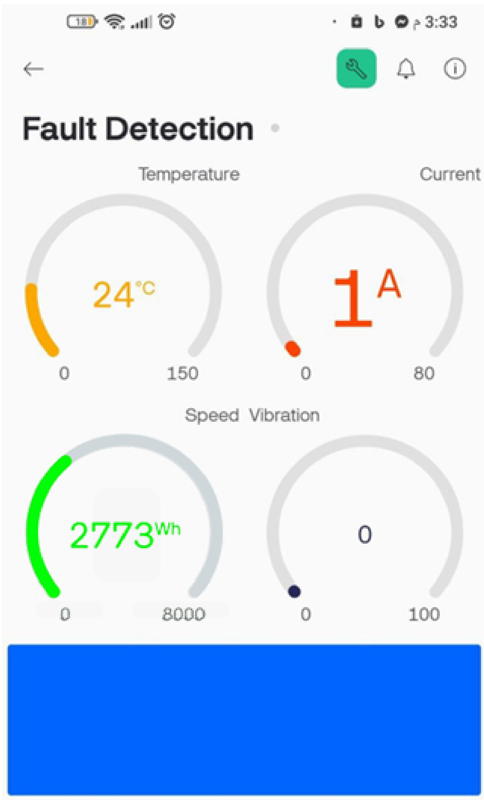

Figure 43: Blynk IoT mobile app dashboard.

12 Results Discussion

The experimental findings illustrate a unambiguous ranking in model performance and convergence behaviors:

the LSTM model alone achieves a significant decline in loss over the 18 epochs but is only able to attain a low, volatile accuracy (15–20%) which suggests the model is learning temporal patterns, yet is unable to convert this learning into consistent classification.

The ANFIS model, by contrast, achieves a stable, linear decline in loss over 50 epochs and identifies some variable relationships to yield a modest validation accuracy of 73% which demonstrates its capacity for capturing nonlinear relationships, but also signifies that its learning features are slower dispersion dynamics relative to the other models.

The hybrid ANFIS+LSTM model was able to leverage the temporal feature extraction performance of LSTM and fuzzy logic inference of ANFIS to converge before 15 epochs to near-perfect training and validation accuracy ($\approx 0.98\%$) without over fitting. All these results highlight that whilst the individual components have their benefits, the hybrid model converges in training more quickly than the previous models and demonstrates powerful predictive performance, which is an important consideration for complex time series classification tasks.

13 Conclusion

In this chapter, we have carefully provided details on our experimental evaluation, presenting a description of the real-world dataset and the preprocessing that made the dataset suitable for fault-classification tasks, reviewing the hybrid architectures (LSTM, ANFIS, and fused architectures), and explaining some of the design decisions in the experiment. Through series of experiments, we demonstrated the constant performance advantageous for the fused ANFIS + LSTM model with precision, recall, and F1-scores nearing 1 across all faults.

The confusion matrices and the loss-accuracy plots show quickly attaining convergence, generalization, and limited over fitting. Overall, this not only validates this methodology but highlights the readiness for industrial application that highlights reliable, consistent, real-time detection of faults which leads to minimized downtime and therefore costs of maintenance within industrial predictive-maintenance contexts.

General conclusion

Deep learning has gained immense traction in both academia and industry due to its brain-inspired ability to learn rich representations for complicated tasks, in applications ranging from remote sensing to intelligent transportation. Deep learning is fundamentally an advanced method that excels at automated knowledge acquisition and automated pattern recognition, and automated decision making which has opened up a wide array of new opportunities for solving challenging problems in artificial intelligence (AI) and beyond.

We leveraged this potential, in this work, to design and develop an effective predictive-maintenance framework for three-phase induction motors. We first described that while an LSTM network was good at discovering temporal patterns, classification accuracy was not consistently reliable with only the LSTM network. We then demonstrated how an ANFIS model (that used fuzzy-logic inference) could also learn nonlinear relationships reliably but converged more slowly. By modelling both the ANFIS and LSTM in a single end-to-end design, we were able to bring together rapid temporal feature learning with interpretable fuzzy rules, achieved fast convergence, close to 98% fault detection performance, and a consistent accuracy rate across all fault types in our predictive-maintenance data-set.

This hybrid model was able to drastically reduce false alarm rates, false negatives, generalized well to unseen data, and provided sufficient flexibility to be a solid foundation for industry grade predictive maintenance models when deployed in the real world.

References

[1] Machine Learning Models for Predictive Maintenance in Industrial Engineering, Magena, Charlene, International Journal of Computing and Engineering, Published: 26th July, 2024.

[2] AI revolutionizing industries worldwide: A comprehensive overview of its diverse applications, Hybrid Advances,Adib Bin Rashid and MD Ashfakul Karim Kausik,2024.

[3] Machine learning in predictive maintenance towards sustainable smart manufacturing in industry 4.0, Zeki Murat and Abdussalam Nuhu, Abubakar and Zeeshan, Qasim and Korhan, Orhan and Asmael, Mohammed and Safaei, Babak, publisher by MDPI,2020.

[4] « ARTIFICIAL INTELLIGENCEr.pdf ».

[5] Machine Learning in the Big Data Age: Advancements, Challenges, and Future Prospects Saeed, Sultan and Ahmed, Suhail and joseph, shalom,2024.

[6] Machine learning and deep learning based methods toward industry 4.0 predictive maintenance in induction motors: State of the art survey, Drakaki, Maria and Karnavas, Yannis L and Tziafettas, Ioannis A and Linardos, Vasilis and Tzionas, Panagiotis, Journal of Industrial Engineering and Management (JIEM),publisher: Barcelona: OmniaScience 2022.

[7] Intelligent fault diagnostic system for rotating machinery based on IoT with cloud computing and artificial intelligence techniques: a review, Maurya, Manisha , Panigrahi, Isham, Dash, Dipti, Malla, Chandrabhanu 2023.

[8] Health monitoring and fault diagnosis in induction motor-a review, Siddiqui, Khadim Moin and Sahay, Kuldeep and Giri, VK, International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 2014.

[9] PREDICTIVE MAINTENANCE IN INDUSTRIAL SYSTEMS USING DATA MINING WITH FUZZY LOGIC SYSTEMS, Selvalakshmi, B and Vijayalakshmi, P and Subha, N and Balamani, T, ICTACT Journal on Soft Computing, 2024

[10] Diagnosis of Multiple Open-Circuit Faults in Three-Phase Induction Machine Drive Systems Based on Bidirectional Long Short-Term Memory Algorithm, Gmati, Badii and Ben Rhouma, Amine and Meddeb, Houda and Khojet El Khil, Sejir, World Electric Vehicle Journal, 2024, publisher by MDPI

[11] ANFIS models for fault detection and isolation in the drive train of a wind turbine, Zemali, Zakaria and Cherroun, Lakhmissi and Hadroug, Nadji and Hafaifa, Ahmed, International Journal of Energetica, 2022

[12] Diagnostic modelling for induction motor faults via ANFIS algorithm and DWT-based feature extraction, Mohamed, Menshawy A and Hassan, Mohamed A Moustafa and Albalawi, Fahad and Ghoneim, Sherif SM and Ali, Ziad M and Dardeer, Mostafa, journal Applied Sciences, 2021, publisher by MDPI

[13] , Harmonics Forecasting of Renewable Energy System using Hybrid Model based on LSTM and ANFIS, Al Hadi, Fawaz M and Aly, Hamed H, journal IEEE Access, 2024.

[14] , A COVID-19 forecasting system for hospital needs using ANFIS and LSTM models: A graphical user interface unit, Shafiekhani, Sajad and Namdar, Peyman and Rafiei, Sima, journal Digital Health, 2022, publisher by SAGE Publications Sage UK: London, England

[15] Design and development of an irrigation monitoring and control system based on blynk internet of things and thingspeak, Saputri, Fahmy Rinanda and Linelson, Ricardo and Salehuddin, Muhammad and Nor, Danial Md and Ahmad, Muhammad Imran, journal PloS one, 2025, publisher by Public Library of Science San Francisco, CA USA

[16] H. D. Shoorkand, M. Nourelfath, and A. Hajji, "Hybrid deep learning for predictive maintenance and production," *Journal of Manufacturing Systems*, vol. 74, pp. 397–410, 2024.

[17] M. L. R. Rodríguez, S. Kubler, A. Giorgio, et al., "Multi-agent deep reinforcement learning for predictive maintenance," *Robotics and Computer-Integrated Manufacturing*, vol. 78, art. 102406, 2022.

[18] S. Elkateb et al., "Machine learning and IoT for predictive maintenance in textiles," *Alexandria Engineering Journal*, vol. 88, pp. 298–309, 2024.

[19] P. Killeen et al., "IoT-based predictive maintenance for fleet management," *Procedia Computer Science*, vol. 151, pp. 607–613, 2019.

[20] W. J. Lee, H. Wu, and H. Yun et al., "AI techniques for predictive maintenance of machine tools," *Procedia CIRP*, vol. 80, pp. 506–511, 2019.

[21] R. G. Goriparthi, "Machine learning for predictive maintenance in industrial IoT," *International Journal of Advanced Engineering Technologies and Innovations*, vol. 1, no. 1, pp. 473–493, 2023.

[22] E. Omol et al., "Anomaly detection in IoT sensor data for predictive maintenance in smart grids," *International Journal of Science, Technology Management*, 2024.

[23] M. Drakaki et al., "Predictive maintenance in induction motors: Industry 4.0 approach," *Journal of Industrial Engineering and Management*, 2022.

[24] S. Kiangala and Z. Wang, “Predictive maintenance for conveyor motors using dual time-series and CNN,” *IEEE Access*, 2020.

[25] F. Olaoye and K. Potter, “Deep learning for predictive maintenance in IoT-connected renewable energy systems,” *Applied Sciences*, 2021.

[26] S. Kim et al., “Fault diagnostics of rolling element bearings using MATLAB envelope analysis,” *Applied Sciences*, 2020.

[27] N. Aissani et al., “Dynamic scheduling of maintenance in the petroleum industry using reinforcement learning,” *Engineering Applications of Artificial Intelligence*, 2009.

[28] L. Cummins, A. Sommers, S. B. Ramezani, S. Mittal, J. Jabour, M. Seale, and S. Rahimi, “Explainable predictive maintenance: A survey of current methods, challenges, and opportunities,” *IEEE Access*, 2024.

[29] R. C. Parpala and R. Iacob, “Application of IoT in predictive maintenance of industrial equipment,” *MATEC Web of Conferences*, 2017.

[30] K. M. Ngu, N. Philip, and S. Sahlan, “Proactive and predictive maintenance strategies for instrumentation control in oil gas,” *International Journal of Integrated Engineering*, 2019.

[31] R. H. Hadi, H. N. Hady, A. M. Hasan, A. Al-Jodah, and A. J. Humaidi, “Improved fault classification for predictive maintenance in industrial IoT,” *Processes*, vol. 11, art. 1507, 2023.

[32] M. R. Nazabadi, S. E. Najafi, A. Mohaghar, and S. Movahedi, “Joint policy of production, maintenance, and product quality in a multi-machine production system,” *Research in Industrial Engineering Journal*, vol. 13, no. 1, pp. 71–87, 2024.

[33] D. D. Jambol, O. O. Sofoluwe, A. Ukato, and O. J. Ochulor, “Transforming equipment management in oil and gas with AI-driven predictive maintenance,” *Computer Science IT Research Journal*, 2024.

[34] M. Kisten, A. E. Ezugwu, and M. S. Olusanya, “Explainable AI model for predictive maintenance in smart agricultural facilities,” *IEEE Access*, 2024.

[35] Z. M. Çınar, A. Abdussalam Nuhu, Q. Zeeshan, O. Korhan, M. Asmael, and B. Safaei, “Machine learning in predictive maintenance towards sustainable smart manufacturing in Industry 4.0,” *Sustainability*, 2020.

[36] S. D. Bristi, M. J. Tatha, M. F. Ali, U. A. Bhatti, S. K. Sarker, M. Masud, Y. Y. Ghadi, A. Algarni, and D. K. Saha, “A meta-heuristic sustainable intelligent IoT framework for bearing fault diagnosis of electric motor under variable load conditions,” *Sustainability*, 2023.

[37] O. Matania, R. Cohen, E. Bechhoefer, and J. Bortman, "Test-training leakage in evaluation of machine learning algorithms for condition-based maintenance," Proceedings of the PHM Society European Conference, 2024.

[38] D. D. Jambol, O. O. Sofoluwe, A. Ukato, and O. J. Ochulor, "Transforming equipment management in oil and gas with AI-driven predictive maintenance," *Computer Science IT Research Journal*, 2024.

[39] N. C. Ohalete and A. O. Aderibigbe et al., "Advancements in predictive maintenance in the oil and gas industry: A review of AI and data science applications," *World Journal of Advanced Research*, 2023.

[40] J. Chukwunweike, I. Abubakar, and A. N. Anang, "Enhancing industrial efficiency through automation: Leveraging PLC, SCADA, HMI, Batch, and DCS systems for downtime mitigation," ResearchGate, 2024.

[41] A. M. Tom and J. L. Febin Daya, "Machine learning techniques for vector control of permanent magnet synchronous motor drives," *Cogent Engineering*, 2024.

[42] M. Al-Greer, I. Bashir, A. Demirel, and O. Keysan, "Autonomous fault detection and diagnosis for permanent magnet synchronous motors using VMD, HHT, and CNN," *Computers and Electrical Engineering*, 2023.