International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Mathematical Modelling of Engineering Problems
Vol. 12, No. 3, March, 2025, pp. 982-990
Journal homepage: http://iieta.org/journals/mmep

Optimizing Wireless Sensor Networks with Machine Learning-Based Predictive Maintenance

Basavaraj Gangasamudra Nagendrappa[1*], Mohan Bangalore Anjaneyalu[1], Sudhamani Makanahalli Veerappa[2], Erathimmana Govindappa Satish[3], Kavitha Chandrasekhar[4]

[1] Department of ISE, BMS Institute of Technology & Management, Visvesvaraya Technological University, Karnataka 560119, India
[2] Department of ISE, B.M.S. College of Engineering, Visvesvaraya Technological University, Karnataka 560019, India
[3] Department of CSE, Nitte Meenakshi Institute of Technology, Bangalore 560064, India
[4] Department of CSE, R.L. Jalappa Institute of Technology, Visvesvaraya Technological University, Karnataka 561203, India

Corresponding Author Email: drbasavarajgn@bmsit.in

https://doi.org/10.18280/mmep.120324

Received: 29 November 2024
Revised: 18 January 2025
Accepted: 24 January 2025
Available online: 31 March 2025

Keywords:
FFNN, Industrial Internet of Things, machine learning, PdM, QoS, WSN

ABSTRACT

Predictive Maintenance systems (PdMs) provide a modern approach for system operators to assess current conditions and predict future system performance, enabling timely maintenance actions. This model integrates quantization and encoding techniques to minimize complexity. Quantization refers to the process of reducing the number of bits used to represent data, while encoding transforms data into a suitable format for processing. The proposed system is applied to Wireless Sensor Networks (WSNs), focusing on predicting data transfer quality. The Feed Forward Neural Network (FFNN), a type of artificial neural network, forecasts the network's functioning status after M steps on basis of earlier Quality of Service (QoS) readings. Our findings indicate that increasing M raises prediction error and complexity, whereas a larger L reduces prediction error but increases complexity. In addition, quantization and encoding effectively lower the system complexity for real-time implementation.

1. INTRODUCTION

PdM system focuses on data collection and system operation estimation. It helps users access system conditions, identify faults, predict failures, and estimate the remaining system life. This paper presents a novel PdM model tailored for WSNs. The main contributions of this work include: The integration of FFNNs to predict network conditions based on multivariate time-series QoS data. The implementation of quantization (reducing the number of bits representing data) and encoding (transforming data into processable formats) techniques to minimize computational complexity. A detailed performance evaluation demonstrating the model's efficacy in real-time deployment scenarios.

These contributions aim to enhance the reliability and efficiency of WSNs while addressing resource constraints. The PdM approach maximizes system lifespan, minimizes unplanned downtimes, and reduces maintenance costs, improving overall reliability and production quality. The WSNs and IoT technologies are essential for advancing and refining Predictive Maintenance. These technologies facilitate the collection of vast amounts of data from sensors installed in machinery, within factories, and at various monitoring locations. For PdM to work effectively, an active sensing system is required to gather measurements that accurately represent the condition of the systems being maintained. The selection, quantity, distribution, and reliability of these sensors are critical factors that determine the overall effectiveness and quality of the PdM.

Most researchers and developers in area of PdM utilize WSNs and IoT as fundamental platform for their solutions. In hazardous and challenging industrial environments, the use of WSNs provides an automated method of performing measuring in lieu of manual measurements. Further, PdM systems are easily deployed and configured due to wireless communication through WSNs. However, these systems may be faced with problems including, energy shortages, force majeure threats, bandwidth issues and restricted computing abilities. Similar to this, PdM systems can be enhanced and optimized with the help of IoT, WSN, machine learning, and deep learning. Typically, ML/DL models, which are developed using neural networks, accept input in two or one-dimensional forms and the outputs are categorical for classification models or a continuous value for regression models.

To achieve high performance and fulfil PdM goals, it appears necessary to choose a design technique that best suits the needs of the system and offers accurate prediction and classification. Based on the ML algorithm, a suggested prediction model framework was created in this regard. It calculates the predictive likelihood distribution of the monitored system's operational status by removing redundant data and streamlining the information into a multivariate time series. It predicts the likelihood that the system will remain fully functional in the upcoming M steps and guarantees that this functionality will be maintained with a specified level of dependability.

The suggested model was created using an ML method based on the feed-forward neural network [1, 2]. Prior observations of the QoS parameters—packet loss, delay, throughput, and power consumption expressed as multivariate time series serve as the input for PdM. After M steps forward in time from the current time, the PdM forecasts the output as a vector that gives the WSN's status. In order to meet the criteria of WSNs, the complexity and memory consumption are further reduced through the use of quantization and encoding techniques [3].

The difficulty of applying PdM in WSNs, where energy and computational resource limitations impair the performance of real-time machine learning models, is addressed in this work. Specifically tailored to the resource constraints of WSNs, the objective is to create a low-complexity, high-accuracy PdM model that can forecast network performance and minimize system outages.

The organization of the paper is as follows: The first section gives introduction to the domain followed by related work. The third section is proposed system model, fourth section presents the overall numerical results and performance of the proposed work in different scenarios, finally the proposed work is concluded.

2. RELATED WORK

In industrial WSNs, the applications are progressively recognizable taking into account the improved flexibility and the cheaper setup costs. But they also create new problems of energy optimization, and network management, among others, which industrial users struggle to solve. To address the challenges, therefore, new forms of optimized energy consumption for Industrial WSNs have been developed by applying Machine Learning techniques [4]. The model also allows for gaining information about possible compromises between power consumption and the speed of communication to achieve a more efficient energy solution. The EEOM achieved an estimated 64.72% transmission energy consumption, 35.28% transmission energy savings, 67.27 percent received energy utilization and 32.73% received energy storage, 52.16% idle mode energy efficiency and 478.4% idle mode energy efficiency, 66.31% sleep mode energy efficiency and 33.69% sleep mode energy storage. The prevalence threshold was 90.44%, 90.06% MCC, 93.93% Delta P, 90.33% CSI, and 90.17% FMI. These automated knowledge-based methods will improve Industrial WSNs' dependability, efficiency, and energy cost savings when combined with a manual intervention [5, 6].

IoT is a special case of Internet of Things that is implemented in production where IoT technologies that bring a wide range of sensors as well as a possibility to use various analytics on a machine data it produces are designed. The data obtained by the use of the machines normally contains date time information which is very important for the predictive models. This paper examines the application of ARIMA forecasting on the time series data that originates from the Slitting Machine sensors, in an endeavor to predict future failures and quality defects. Machine Learning thus establishes as a fundamental requirement in IoT owning applications in quality assurance and quality checking, reducing the cost of manufacturing and boosting the general productivity of production [7].

They say that the mere prediction of the weather has the potential to save the many lives in environmental disasters such as landslides, earthquake, flood, forest fire, tsunamis, and so on by disaster monitoring and people warning them to evacuate in the event of the occurrence of disaster. In this paper, we use Multiple Linear Regression (MLR) model for humidity prediction. Finally, the Multiple Linear Regression technique was used to predict the humidity after exploratory data analysis and outlier treatment was done. Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE) have been used as the performance criteria to assess the model. From this experiment the applied method produced results with minimum error of 11% thus the model is significant and predictions more accurate than other methods [8].

The deployment of Deep Learning-based Predictive Maintenance (DL-PM) in Industrial Internet of Things (IIoT) enhances productive performance and minimizes the availability of idle time. This research work looks at the integration of deep learning with the Predictive Maintenance in the Industrial IoT domain. The research identifies the shift from orthodox maintenance to intelligent approaches such as deep learning made possible by IoT data capture in real time. In methodological terms, it presents data preprocessing, model selection and design for DL-PM. CNNs process data captured by sensors, RNNs forecast temporal patterns, and hybrid models incorporate transfer learning [9]. The research work presents the use of DL-PM across industries and industries using quantitative measurements and qualitative assessments that compare DL-PM to traditional methods. It covers issues such as data quality, model types explain ability, model scalability, model sensitivity, and accessibility to ethical issues. This research work increases appreciation of the transformative impact of DL-PM in IIoT. The ability of DL-PM to change industrial process in the IoT era is highlighted. It outlines the approach to implementing DL-PM, the issues that may be faced and the opportunities for industries to develop effective approaches for operation that is not disrupted by frequent downtime periods [10].

Industry 4.0 has provided vital inputs for the propulsion of the automotive sector which engulfs various areas of economic and social development and the corporate models of smart production, smart manufacturing, and IoT, IoT were also propelled by Industry 4.0. This paper presents a basic assessment of trends on machine learning techniques that are more generally applicable to PdM in smart manufacturing in accordance with Industry 4.0. The study adopts the classification of research based on ML algorithms and deals with the future prediction of a temperature using time series and multivariate analysis methods. Through the presentation of the recent developments in ML and PdM of smart manufacturing, this paper seeks to advance the knowledge in improving the manufacturing processes for the advancement of competitiveness in the manufacturing sector [11, 12].

According to the previously described development, the PdM is a new technology that assists system operators in assessing the systems' existing condition as well as forecasting their future quality and maintenance action management. This study creates a work flow PdM model that employs a machine learning technique to ascertain the operational state of the system following M action steps from the L actions of a FFNN. In order to simplify the system, we employ quantization and encoding techniques to cast the problem in the discrete time domain in order to foresee the system's condition in terms of data quality transmission, we use the suggested model in this study to build a PdM system within WSNs. By examining the prior L records of QoS measures from the WSN, the FFNN architecture enables the projection of the network's operational state after M consecutive future time steps. We have discovered that an increase in M leads to greater complexity and a higher prediction error, while a larger L is linked to increased complexity but a reduction in prediction error. Furthermore, we investigate how quantization and encoding techniques can aid in reducing complexity to achieve a real-time PdM system [13].

Internet of things (IoT) is basically a platform which deals with management of daily life activities to create an interaction between things and people. Among them, an example of the application of a smart office that connects electrical appliances and sensors via the Internet using an automation system is described in this paper. The data collected from different sensors and appliances go to the cloud, and the data is available to the user through the smartphone no matter their location. In this paper a sensor fault prediction model using a machine learning algorithm is established where ‘k-nearest neighbors model’ outperforms other models with an accuracy of 99.63%, F1- score of 99.59% and recall of 99.67%. In an attempt to assess the performance of the above models, i.e. k-nearest neighbors and Naive Bayes, several performance parameters including precision, recall, F-measure, and accuracy were used. It is a safe, uninterrupted, and stable automated system that makes smart office employees and their work safe and efficient and saves resources [14]. Sudden failure of the important equipment affects the overall production line rate of the IIoT and to achieve a higher leading-edge business acumen, the PdM works on working data. Employers on the other hand are always under pressure to manually route suitably competent manpower resources wherever there is a machine breakdown.

Moreover, for equipment, human error is found to have a ripple effect in terms of negative impact on total equipment availability time and the production timeline. Hence, in this paper, the complex resource management problem is cast into a resource optimization problem to know if a model-free deep reinforcement learning (DRL) based PdM framework can learn the optimal decision policy from the stochastic environment automatically. In contrast to the proposed PdM frameworks, our approach takes into account the information from health sensors of PdM and resources of physical equipment as well as human into account the optimization problem [15].

Global pressures to increase value addition and reduce costs make oil and gas refinery operations a complex task. Predictive Maintenance strategies have become the critical solution through this approach to carry out real-time abnormality detection, pressure variation forecast, and asset condition tracking. A good and informative case is a downstream asset established in Western Australia that skillfully exploits the potential of the IoT and AI/ML. Technical focuses include using wireless sensors for data capture, data transfer to a central control, and the application of machine learning to recognize equipment aberrations. Experts and decision makers receive these defects at the earliest and with contextual information they are in a position to make sound decisions quickly [16].

Maintenance which is performed or not done in correct manners affect that mean time between failure (MTBF) to worsen. Manual diagnostic procedures normally lead to an increase in the time taken to repair the system at the point of breakdown. Working component starts with attribute selection with a correlation ranking filter to determine the most critical attribute to system condition. In the subsequent step, ensemble learning procedures were specified to develop a PdM model. Wait using a numeric data experiment determined six of the wafer stick machine attributes affecting system condition and a working test on the proposed PdM model yielded 95.90% accuracy [17].

Due to a large number of devices integrated into smart grid infrastructure as part of the IoT, large amounts of sensor data can be produced. Such a wealth of data offers a signal that advanced data analysis approaches to PdM can be applied to smart grids. This paper also presents a discussion on the performance evaluation and model selection for anomaly detection in IoT sensor data and the integration and deployment possibilities and limitations and critique of the few selected empirical works. Interestingly, the study asks for the barriers that could occur in the course of executing and implementing Machine Learning Techniques particularly in Anomaly Detection in IoT Sensor Data. Further, the study makes a contribution to the features that might have possibly been left unnoticed in the preexisting literature [18].

The industry of marine transport is gradually moving towards the implementation of condition-based maintenance for the enhancement of dependability and efficiency of marine propulsion systems. This work presents an exploratory study on marine propulsion health monitoring using neural networks and IoT sensor fusion. It is gathered in real time by a complex array of sensors covering the essential parts of the propulsion systems. Combining sensor data via state-of-art IoT algorithms provides a holistic insight into the system’s condition. The technique proposed in the current work focuses on the Predictive Maintenance using neural networks. Neural networks and IoT sensor fusion provide prospects for early defect detection and real-time schedules for equipment maintenance [19].

The probability of incurring additional operative cost due to equipment failure in the course of its productive use might be reduced by regularly scheduling maintenance check on revenue generating assets to identify, prevent and rectify potential faults before they result in fatal breakdowns. In each of the equipment, raw sensor data are collected for a sensor device and analyzed for equipment health status to identify anomalous events. “As for the black-box regression models, the proposed algorithm learns an optimal maintenance policy and offers a recommendation for each equipment” [20].

In relation to IoT, this paper describes possibilities to increase the security of WSNs. Notably it responds to the issues brought by non-homogenization of IoT. On the other hand, this paper also uses the machine learning and AI algorithms in the detection of the intrusion detection. WSNs, which play a significant role in acquiring data through multiple applications, will be vulnerable to threats such as wiretapping and DoS attack. This method is better at authentication efficiency as well as, better at improving the detection and segregation of multiple types of security threats for opening up more possibilities for innovation within the cyber security of IoT [21].

Scalability and real-time processing are issues for many models, though, because of computing limitations. Our method, which combines quantization and encoding techniques with a FFNN, offers a unique balance between prediction accuracy and reduced model complexity for real-time deployment in WSNs with restricted resources. By addressing resource efficiency and scalability, this methodology improves PdM efficacy in WSN settings.

3. PROPOSED SYSTEM MODEL

There are two basic parts to the proposed PdM approach:

(i) Forecasting model for the forward probability distribution of the system's operating state under observation. Time-series data is used to display the system information, and the model forecasts how the system will operate over the following M steps to ensure dependable system operation with a specified reliability level parameter.

(ii) To enhance the prediction model, an ML algorithm that has been employed represents the prediction model, that is, a FFNN.

3.1 Estimating future probability distribution

Assume that the system information is represented by time series $w(t)$, which can be derived from direct measurement or processed data. The system status is evaluated as follows:

• If $w(t)\leq D$, then the system is considered faulty and urgent repairs are necessary.

• If $w(t)\leq D$, the system functions normally.

The goal is to estimate likelihood that the system will remain operational over the next N time steps, given past observations $w(t-1)$, $w(t-2)$, ..., $w(t-L+1)$. This probability can be represented as:

$$P(w(t+N)\leq D, w(t+N-1)\leq D, \ldots, w(t)\leq D) \\ (w(t-1)=u, \ldots, w(t-L+1)=v) \tag{1}$$

More precisely, the objective is to determine whether the system's continued operability over the next N time steps is assured with a specified reliability level, determined by the parameter δ, as shown in:

$$P(w(t+N)\leq D, w(t+N-1)\leq D, \ldots, w(t)\leq D) \\ (w(t-1)=u, \ldots, w(t-L+1)=v) \geq 1-\delta \tag{2}$$

Define the following notations for convenience:

$$Y^{+(t)} := \left(y(t+N), y(t+N-1) \ldots\ldots.., y(t)\right) \\ y^{\wedge}-(t) := (y(t-1), \ldots, y(t-L+1)) \tag{3}$$

In this form, the probability can be expressed more compactly, where the set C is defined as:

$$C := \{\, y_i < D, \ldots., y_{\{N\}} \ \leq D \ for\ 1 \leq I \leq \ N \,\}$$

Thus, the probability is given by:

$$P\left(y^{+(t)} \in C \,\middle|\, y^{-(t)} = (u, \ldots, v) \right) \geq 1-\delta \tag{4}$$

Define the following two vectors:

$$r(1) = \left(\{r\}_1^{\{(1)\}}, \{r\}_2^{\{(1)\}}\right) = (1,0)\{\ indicating\ \}y^{+} \in C \\ r(2) = \left(\{r\}_1^{\{(2)\}}, \{r\}_2^{\{(2)\}}\right) = (0,1)\{\ indicating\ \}y^{+} \notin C \tag{5}$$

This allows us to create a training set:

$$\tau(K) = \{(y^{\wedge}(t), r(t)) \mid t = 1, \ldots, K\,\}, \\ r(t) \in \{r(1),\backslash r(2)\} \tag{6}$$

3.2 FFNN algorithm

The proposed task is implemented using the FNN machine learning technique. The basic architecture of FNNs, which consists of several hidden layers, input layers, and output layers, is what defines them. As the number of hidden layers increases, the data flows unidirectional from the input layer to the output layer. Back propagation, a basic and popular technique, is used as the training method in this study. An activation function is used to determine the estimated output, and a loss function is used to calculate the estimation error. Back propagation is used to update the weights according to the gradient of the loss function.

Several factors influence FNN performance, including the size of the training set, the training algorithm, the structure of the hidden layers, the activation function, and a detailed problem description. The primary criterion is user satisfaction with regard to accuracy and complexity because there are no established standards for choosing, contrasting, and assessing solutions.

The training dataset described in Eq. (6) is used to train the FNN, where the mapping of inputs to outputs is given by $z=Net(u,v)$ with v representing the weights to be learned. The weights adjustment/optimized done using back propagation algorithm are as follows:

$$v_{\{\{opt\}\}} = min_{\{v\}} \frac{1}{k} \sum_{\{k=1\}}^{\{K\}} |t^{\{(k)\}} - \{Net\}\left(u^{\{(k)\}}, v\right)|^2 \tag{7}$$

$$\frac{1}{k} \sum_{\{k=1\}}^{\{K\}} |t^{\{(k)\}} - \{Net\}\left(u^{\{(k)\}}, v\right)|^2 \\ \to E|t - \{Net\}(u,v)|^2 \tag{8}$$

$$v_{\{\{opt\}\}} = min_{\{v\}E}|t - \{Net\}(u,v)|^2 \\ \to \{Net\}(u,v) = E(\,t \mid u\,) \tag{9}$$

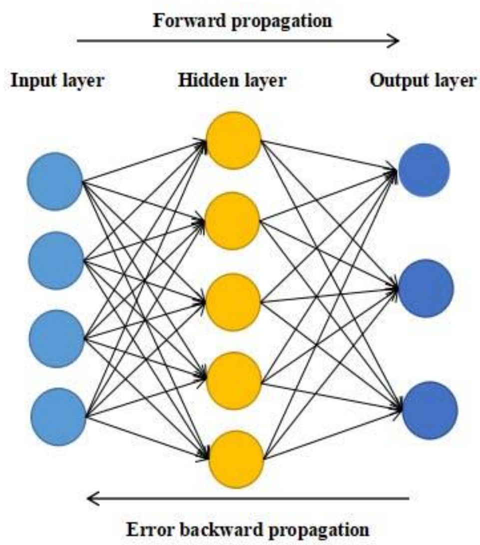

Figure 1. Architecture of a FFNN

After training, the output of the FNN provides the estimated conditional probabilities based on the given past observations. If $P(w^{+} \in C/u) \geq 1-\delta$, then there are at least N steps to failure.

The architecture of a FFNN is depicted in Figure 1, consisting of three hidden layers. The input layer takes in a window of previous observations. These observations are processed through the hidden layers, and the data flows unidirectional from the input layer to output layer. Each hidden layer applies an activation function, and the output layer generates the final prediction or probability estimation.

The architecture includes:

·Input Layer: Accepts a window of past observations (L previous QoS readings).

·Hidden Layers: The network comprises three hidden layers:

·First hidden layer: 128 neurons with ReLU (Rectified Linear Unit) activation function.

·Second hidden layer: 64 neurons with ReLU activation function.

·Third hidden layer: 32 neurons with ReLU activation function. Dropout layers are incorporated after each hidden layer to prevent overfitting, with a dropout rate of 0.3.

·Output Layer: A single neuron with a sigmoid activation function to output the probability of the system's operational status.

The FFNN uses back propagation as the training algorithm, optimizing weights based on the gradient of the loss function (binary cross-entropy for classification tasks). Batch normalization is applied to stabilize learning and speed up convergence. This architecture effectively balances model complexity and computational efficiency, making it suitable for deployment in resource-constrained WSN environments.

3.3 Customized PdM for WSNs

The proposed paradigm is covered in this part as a PdM strategy for WSNs. A WSN is made up of one or more base stations and several tiny nodes that work together to collect data. The nodes use a wireless radio transceiver to connect with the base station and with each other. Small-scale processing units and sensors to measure or track physical phenomena are included with these nodes. WSN nodes are frequently installed in complicated situations and typically run on batteries with a limited power source. Restricted memory, low processing power, limited communication bandwidth, and inadequate power backup are some of the constraints that WSN designers and operators must take into account. Notwithstanding these limitations, the system's functionality must satisfy specific QoS standards. Dependability, energy efficiency, security, accuracy, and low latency are some of the QoS requirements for WSNs. In order to enhance WSN performance, the maintenance action may entail selecting new cluster heads, rearranging clusters, installing new sensors, controlling ON/OFF schedules, and other tactics. A low-complexity PdM model is required due to the restricted resources of WSNs.

3.4 Quantized FFNN

The energy, memory, and computing capability of WSNs are limited. A quantization approach that speeds up training and lowers the model's complexity is used to make sure the PdM model performs effectively under these limitations. Quantization speeds up training and makes the model simpler, but it may also decrease accuracy, necessitating a trade-off between precision and complexity. The quantization function converts variables and weights from their typical floating-point representations into integers, which are fixed-point representations of numbers. This conversion improves memory efficiency and computation speed.

The quantization level q of the actual value v is determined by a deterministic quantization function in the manner described below:

$$q(v) = \{sign\}(v) \cdot \Delta(\{|v|\}/\{\Delta\} + \{1\}/\{2\}) \qquad (10)$$

where, Δ is the resolution or the quantization step size.

We refer to these functions as equidistant quantization. Each quantization level has an equal portion of the quantization range. Data that is uniformly distributed is ideal for these functions. Non-equidistant quantization works better for distributions that are not uniform.

This algorithm minimizes mean square quantization error σ by taking the probability distribution function (PDF) of the samples into account.

The optimal quantized level $q_j(v)$ of a sample v is found iteratively:

$$q_{j(v)} = \frac{\left\{\int_{\{d_j\}}^{\{d_{\{j+1\}}\}} v \cdot h(v), dv\right\}}{\left\{\int_{\{d_j\}}^{\{d_{\{j+1\}}\},} h(v)\, dv\right\}} \qquad (11)$$

The PDF of the sample distribution is denoted by $h(v)$, whereas d_j and d_{j+1} stand for the limits of the suggested quantization level q_j.

The goal is to minimize the mean square quantization error τ, expressed as:

$$\tau = \int_{\{d_j\}}^{\{d_{\{j+1\}}\}} \left(v - q_j\right)^2 h(v)\, dv \qquad (12)$$

By using this approach, the quantization error is minimized while considering the underlying distribution of the data. Quantization and encoding techniques have a significant impact on the model's performance and complexity. Quantization reduces the model's memory footprint, making it suitable for deployment on devices with limited resources. Fixed-point operations resulting from quantization are computationally faster than floating-point operations, reducing inference time. While quantization may introduce minor inaccuracies, these are offset by the model's robustness, which is particularly beneficial in applications where precision is not critical. Furthermore, dropout layers and encoding techniques enhance the model's ability to generalize, ensuring reliable predictions even with noisy or incomplete data.

Quantization reduces the model's memory and computational requirements by mapping floating-point parameters to fixed-point representations. While this optimization enables deployment on resource-constrained devices, it introduces trade-offs. Specifically, lower precision may slightly degrade model accuracy. However, these inaccuracies are typically negligible in WSN applications where computational efficiency and energy savings are paramount. The benefits include reduced memory footprint, faster inference times, and improved suitability for real-time processing in WSN environments.

3.5 Sparsity of FFNN

The main issue with using FFNNs on WSNs is memory limitations. A number of techniques are used to advance ML/DL algorithms' memory efficiency. While some strategies target memory demands during training, others concentrate on reducing the amount of memory needed for inference. Sparse FFNNs are one popular and effective method for enhancing DL/ML algorithms.

A sparse vector, with the majority of its entries being zeros, is used to represent input characteristics in a sparse FFNN. As a result, less memory and computational load are required. Sparsity can have a detrimental effect on the FFNN's accuracy even while it lowers computational complexity and increases memory economy. As a result, the degree of sparsity and the model's accuracy need to be balanced. We use a straightforward encoding system in this work, which was motivated by the approach described in the previous study. The quantization approach, which encodes each quantized level into an orthonormal vector, is enhanced by this encoding technique:

$$q_m \rightarrow t_q^m : t_q^m(j) = \left\{ \begin{matrix} 1 \quad if \quad j = m \\ 0 \quad otherwise \end{matrix}, j = \{1,2 \dots \dots T\} \right\} \tag{13}$$

By applying this encoding, Eq. (3) becomes:

$$\begin{aligned} z^+(n) &:= \left(t_q(n+P), t_q(n+P-1), \dots . t_q(n)\right), \\ z^-(n) &:= (t_q(n-1), \dots . t_q(n-K+1)) \end{aligned} \tag{14}$$

3.6 Dataset setup

In order to gather data, two TelosB nodes were connected via an IEEE 802.15.4 link that was constructed on TinyOS. Every node was fitted with a 250 kbps TI CC2420 radio transceiver. They tracked packet delivery performance based on several predefined parameters associated with the physical, MAC, and application layers. A table of 10,000 observations was generated. Each row in the table summarizes the average values of 300 packets.

The transmission power was fixed at -19 dBm, while other parameters were varied according to the combinations listed in Table 1.

In addition to these configured parameters, the table also records a number of performance metrics related to packet delivery for each parameter combination, as presented in Table 2. A trial of the observation is provided in Table 3. The table provides shows configured parameters for a system, along with their acronyms, possible values, and comments. Here's a breakdown:

Table 1. Parameters configuration

Parameter Name	Abbreviation	Possible Values	Notes
Time Between Arrivals	TBA	10, 20, 25, 30, 35, 40, 45, 50 ms	Pre-set
Data Packet Size	DPS	25, 35, 55, 70, 85, 100, 110 bytes	Pre-set
Maximum Queue Depth	MQD	5, 25, 60	Pre-set
Max Retransmission Attempts	MRA	2, 4, 6	Pre-set
Retransmission Delay	RD	25, 55 ms	Pre-set
Signal Transmission Power	STP	18 dBm	Fixed
Node Separation Distance	NSD	12, 22, 32 m	Pre-set

Table 2. Performance metrics

Parameter Name	Abbreviation	Modified Values	Notes
Queue Depth Measurement	QDM	real values (0–60)	Collected
Buffer Overflow Status	BOS	real values (0–1)	Collected
Transmit Attempt Count	TAC	real values (0–5)	Collected
Acknowledged Transmission	ATX	-	Collected
Signal Strength Measurement	SSM	-	Collected
Noise Level	NL	-	Collected
Signal Quality Index	SQI	-	Collected
Time of Packet Arrival	TPA	-	Collected

Table 3. Observation values

Parameter Name	Abbreviation	Values	Notes
Time of Packet Arrival	TPA	125300, 130750, 137710, 146180, 156150	Measured
Time Between Arrivals	TBA	12, 18, 12, 18, 52	Pre-defined
Data Packet Size	DPS	25, 40, 60, 90, 115	Pre-defined
Queue Depth	QD	2, 2, 28, 2, 58	Measured
Retransmission Attempts	RTA	2, 2, 6, 2, 6	Measured
Retransmission Delay	RD	28, 28, 28, 58, 58	Measured
Transmission Power	TP	18, 18, 18, 18, 18	Fixed
Distance Between Nodes	DBN	12, 12, 12, 22, 34	Pre-defined
Buffer Overflow Status	BOS	0, 0, 0, 0, 0	Measured
Queue Value	QV	0.43, 0.25, 26.5, 0.03, 0.10	Measured
Acknowledgement Rate	AR	0.61, 0.80, 0.73, 1.01, 1.03	Measured
Transmission Attempt Count	TAC	0.605, 0.80, 0.73, 1.01, 1.05	Measured
Signal Strength Measurement	SSM	7.50, 9.85, -9.25, -16.30, 22.95	Measured
Noise Level	NL	54.00, -70.50, -61.05, 88.90, -93.70	Measured
Signal Quality Index	SQI	64.00, 83.34, 78.28, 107.10, 106.10	Measured

We calculated the QoS requirements of the WSN based on both pre-set and observed parameters, focusing on energy efficiency, throughput, delay, and packet loss.

The reliability of the system is reflected in the packet error rate (PER), which is affected by the nodes' queuing behavior and the quality of connection metrics like LQI, RSSI and NF.

PER is calculated as:

$$PER = \frac{\{N_{\{p\}} - A\}}{\{N_{\{p\}}\}} \tag{15}$$

Energy efficiency (E_{eff}): This measures the energy required to transmit a useful bit. It is influenced by *PER*, transmission power level, packet payload, header length, and transmission rate:

$$E_{\{eff\}} = \frac{\{P_{\{tx\}} \times (H_{\{L\}} + P_{\{L\}}) \times T_{\{s\}}\}}{\{P_{\{L\}(1-\{PER\})}\}} \tag{16}$$

where, T_s is the transmission time, which is 0.004 ms at a rate of 250 kb/s, and H_L is the header length, which can vary between 11 and 31 bytes in IEEE 802.15.4.

Throughput (T_p) refers to the number of useful bits received per unit time. It determined by on packet payload (P_L), PER, and transmission service time (T_{srv}):

$$\begin{gathered} Throughput\ (T_p) = \frac{\{P_{\{L\}(1-\{PER\})}\}}{\{T_{\{srv\}}\}} \\ (T_{srv}) = K + T_{\{s\}} + (N_{\{p\}} \times D_{\{R\}}) \\ Transmission\ Service\ Time\ (T_{srv}) \end{gathered} \tag{17}$$

where, K is a constant depending on the protocol and radio system specifications. In this case, it is approximately 13.5 ms based on the experimental setup.

Delay refers to the time from packet creation to its successful reception. It is influenced by the link quality indicator (LQI) and node queuing characteristics. Often, a queuing model is used to represent delay in WSNs. We use system utilization (u) as a metric for delay, where $u = T_{srv}/TBA$, and as $u \rightarrow 1$ delay increases.

The computed QoS metrics (*PER*, E_{eff}, T_p, and u) are organized into a 10,000×4 input feature table, where each row corresponds to an entry in the observations table. As a fifth input feature, the time of packet arrival (TPA) is converted into a time series and added. The QoS metrics frequently depend on one another; for example, by increasing energy efficiency, throughput may decrease, or dependability may increase. For the user, these metrics must be balanced. The following ranges are established for each statistic in order to specify the WSN's operating conditions:

$$\alpha^{\{+\}} \leq \{PER\} < \alpha^{\{-\}},\ \beta^{\{+\}} \leq E_{\{eff\}} < \beta^{\{-\}},\ \gamma^{\{+\}} \leq T_{\{p\}} < \gamma^{\{-\}},\ \delta^{\{+\}} \leq u < \delta^{\{-\}} \tag{18}$$

4. RESULTS AND DISCUSSIONS

The trends indicate that both packet loss and delay rise with increased network traffic, while throughput grows initially but reaches a cap due to bandwidth constraints. These patterns highlight points where Predictive Maintenance could effectively address performance declines. Figure 2 shows changes in essential QoS metrics (e.g., packet loss, delay, and throughput) over time, illustrating how the network reacts to varying conditions.

Accuracy tends to drop with larger M values, reflecting the increasing challenge of predicting farther into the future. This emphasizes a balance between forecast range and precision, indicating that setting optimal M values could enhance the model's practicality for PdM use. Figure 3 shows how prediction accuracy changes as prediction steps (M) are extended.

By incorporating more past observations, the model's reliability improves. However, diminishing returns at higher L values suggest there's an optimal historical data size that balances improved accuracy with computational efficiency—helpful for WSN PdM applications. Figure 4 demonstrates that prediction error decreases as the amount of historical data (L) included in the model increases.

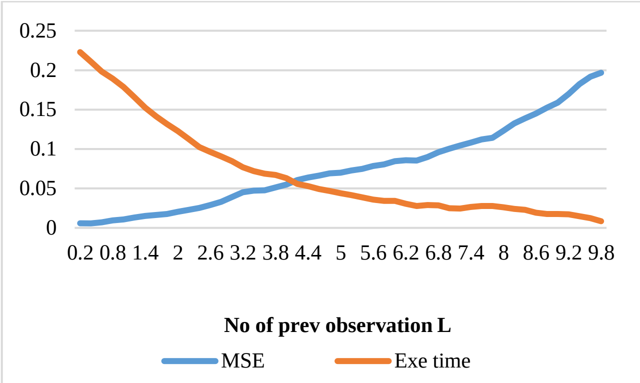

Figure 2. Relation between MSE and execution time

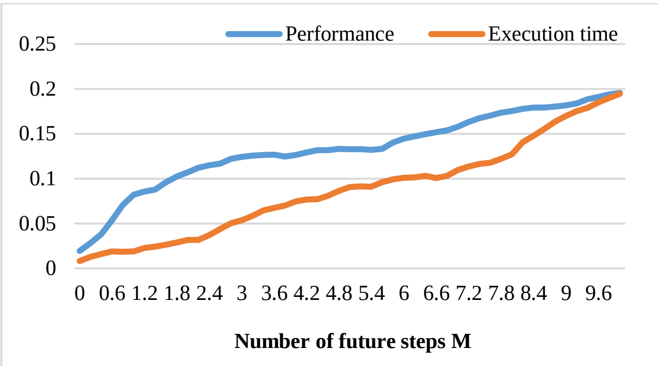

Figure 3. Prediction accuracy with increased prediction steps (M)

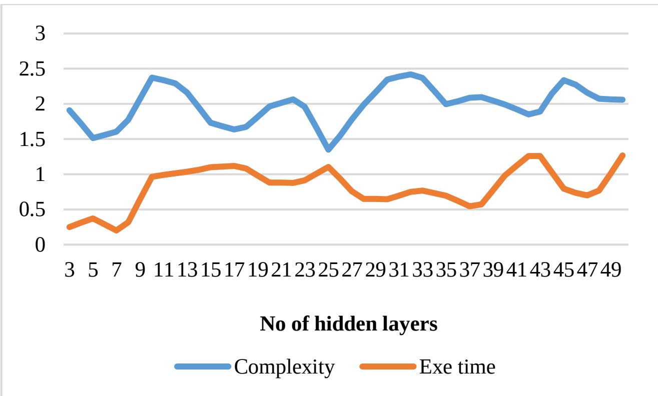

Figure 4. Historical data size (L) and prediction error

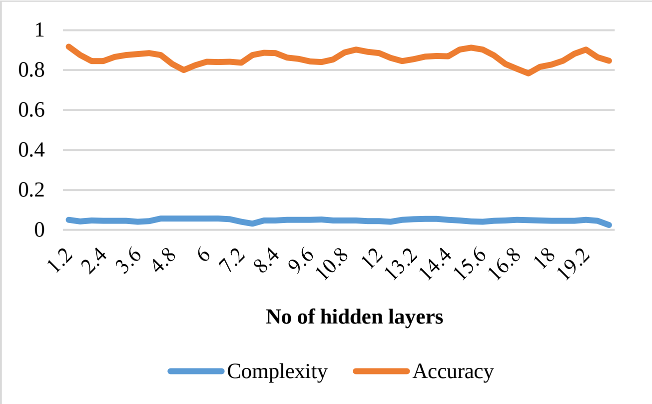

Figure 5. Quantization levels and model complexity

The results stress the importance of choosing an appropriate quantization level to maintain efficient operation in WSNs without significantly impacting prediction accuracy.

Figure 5 presents the trade-off between quantization levels and model complexity, where higher levels of quantization reduce computational load but may introduce minor inaccuracies.

The proposed model was compared with benchmark PdM models, including LSTM, GRU, and SVM-based approaches. While LSTM and GRU excelled at capturing temporal dependencies, their high computational demands made them unsuitable for resource-constrained WSNs. SVM models, though efficient, struggled with handling multivariate QoS data. The proposed FFNN, with quantization and encoding techniques, achieved comparable or superior accuracy while maintaining lower memory and computational requirements, demonstrating its practicality for WSN deployment.

During the analysis, some anomalies were observed in the data, such as sudden spikes in QoS metrics like delay and packet loss, which occurred under specific environmental conditions, including extreme temperature fluctuations. These anomalies suggest potential limitations of the sensors or the communication protocol under adverse conditions.

Statistical significance testing and 95% confidence intervals were used to validate the model's reliability. The impact of quantization on model performance was evaluated. Results indicate that quantization reduced memory usage by 87% and computation time by 86%, with more accuracy. These trade-offs are acceptable for WSNs, as the system prioritizes efficiency over marginal gains in precision.

4.1 Evaluation metrics

The proposed model was evaluated using accuracy, precision, recall, F1-score, Mean Absolute Error (MAE), and execution time. Accuracy ensures overall reliability, while precision and recall focus on minimizing false positives and capturing all faults, respectively, critical for WSN maintenance. The F1-score balances precision and recall for imbalanced datasets, and MAE quantifies QoS prediction errors for system optimization. Execution time ensures the model meets real-time requirements, addressing the efficiency and resource constraints of WSN environments.

5. CONCLUSIONS

This paper introduces a machine learning-driven approach to PdM tailored for WSNs. Using a FFNN, the model forecasts future network conditions by analyzing QoS metrics. By incorporating quantization and encoding, the model reduces complexity, making it suitable for real-time deployment in resource-limited WSNs. Findings reveal that longer prediction steps (M) can increase complexity and error, while a larger historical data window (L) improves accuracy. Overall, the PdM model achieves a practical balance between prediction precision and resource limitations, supporting reliable network performance with efficient energy use and minimized packet loss. The limitations include its reliance on a fixed data window size and FFNN, which may not generalize well to other network types or capture long-term dependencies.

Deploying the model in real-world WSNs involves addressing challenges like resource constraints, environmental variability, and network latency. Quantization and encoding techniques ensure efficient operation on resource-limited devices, while edge computing mitigates latency by enabling local data processing. For diverse applications, the model can be optimized by tailoring input features and retraining with domain-specific datasets.

REFERENCES

[1] Gulyani, V., Dhiman, T., Bhushan, B. (2020). Introducing machine learning to Wireless Sensor Networks: Requirements and applications. In Deep Learning Strategies for Security Enhancement in Wireless Sensor Networks, pp. 1-22. https://doi.org/10.4018/978-1-7998-5068-7.ch001

[2] Almazaideh, M., Levendovszky, J. (2022). A Predictive Maintenance system for Wireless Sensor Networks: A machine learning approach. Indonesian Journal of Electrical Engineering and Computer Science, 25(2): 1047-1058. https://doi.org/10.11591/ijeecs.v25.i2.pp1047-1058

[3] Vlasov, A.I., Grigoriev, P.V., Krivoshein, A.I., Shakhnov, V.A., Filin, S.S., Migalin, V.S. (2018). Smart management of technologies: Predictive Maintenance of industrial equipment using Wireless Sensor Networks. Entrepreneurship and Sustainability Issues, 6(2): 489-502. https://doi.org/10.9770/jesi.2018.6.2(2)

[4] Bagwari, A., Logeshwaran, J., Usha, K., Kannadasan, R., Alsharif, M. H., Uthansakul, P., Uthansakul, M. (2023). An enhanced energy optimization model for industrial Wireless Sensor Networks using machine learning. IEEE Access, 2023: 3311854. https://doi.org/10.1109/access.2023.3311854

[5] Kumar, V., Prakash, M., Thamburaj, S. (2024). Deep learning-based Predictive Maintenance for industrial IoT applications. In 2024 International Conference on Inventive Computation Technologies (ICICT), pp. 1197-1202. https://doi.org/10.1109/icict60155.2024.10544377

[6] Mohan, B.A., Sarojadevi, H. (2016). A hybrid approach for data collection using multiple mobile nodes in WSN (HADMMN). In 2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), pp. 711-714. https://doi.org/10.1109/RTEICT.2016.7807916

[7] Rai, A.K., Pokhariya, H.S., Tiwari, K., Vani, V.D., Kumar, D., Kumar, A., Rana, A. (2023). IoT driven Predictive Maintenance using machine learning algorithms. In 2023 6th International Conference on Contemporary Computing and Informatics (IC3I), pp.

2674-2678. https://doi.org/10.1109/ic3i59117.2023.10398108
[8] Serradilla, O., Zugasti, E., Rodriguez, J., Zurutuza, U. (2022). Deep learning models for Predictive Maintenance: A survey, comparison, challenges and prospects. Applied Intelligence, 52(10): 10934-10964. https://doi.org/10.1007/s10489-021-03004-y
[9] Karthik, S.A., Basavaraj, G.N., Seemanthini, K., Kumar, H., Supriya, R.K. (2023). Hybrid approach for retail store auditing using CRNN. In 2023 International Conference on Network, Multimedia and Information Technology (NMITCON), pp. 1-7. https://doi.org/10.1109/NMITCON58196.2023. 10276301
[10] Wang, H., Zhang, W., Yang, D., Xiang, Y. (2022). Deep-learning-enabled predictive maintenance in industrial Internet of Things: Methods, applications, and challenges. IEEE Systems Journal, 17(2): 2602-2615. https://doi.org/10.1109/JSYST.2022.3193200
[11] Sharma, M., Sharma, M., Sharma, M., Yadav, K.S., Shukla, S. (2023). Industry 4.0 technologies for smart manufacturing: A systematic review of machine learning methods for Predictive Maintenance. In 2023 International Conference on Self Sustainable Artificial Intelligence Systems (ICSSAS). https://doi.org/10.1109/icssas57918.2023.10331740
[12] Basavaraj, G.N., Lavanya, K., Reddy, Y.S., Rao, B.S. (2022). Reliability-driven time series data analysis in multiple-level deep Learning methods utilizing soft computing methods. Measurement: Sensors, 24: 100501. https://doi.org/10.1016/j.measen.2022.100501
[13] Kanawaday, A., Sane, A. (2017). Machine learning for Predictive Maintenance of industrial machines using IoT sensor data. In 2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS), pp. 87-90. https://doi.org/10.1109/icsess.2017.8342870
[14] Uppal, M., Gupta, D., Goyal, N., Imoize, A.L., Kumar, A., Ojo, S., Pani, S.K., Kim, Y., Choi, J. (2023). A real-time data monitoring framework for predictive maintenance based on the internet of things. Complexity, 2023(1): 9991029. https://doi.org/10.1155/2023/9991029
[15] Ong, K.S.H., Wang, W., Niyato, D., Friedrichs, T. (2021). Deep-reinforcement-learning-based predictive maintenance model for effective resource management in industrial IoT. IEEE Internet of Things Journal, 9(7): 5173-5188. https://doi.org/10.1109/jiot.2021.3109955
[16] Saboo, S., Shekhawat, D. (2024). Enhancing Predictive Maintenance in an oil & gas refinery using IoT, AI & ML: An generative AI solution. In International Petroleum Technology Conference, p. D031S128R003. https://doi.org/10.2523/iptc-23466-ms
[17] Mujib, A., Djatna, T. (2020). Ensemble learning for predictive maintenance on wafer stick machine using IoT sensor data. In 2020 International Conference on Computer Science and Its Application in Agriculture (ICOSICA), pp. 1-5. https://doi.org/10.1109/icosica49951.2020.9243180
[18] Omol, E., Mburu, L., Onyango, D. (2024). Anomaly detection in IoT sensor data using machine learning techniques for predictive maintenance in smart grids. International Journal of Science, Technology & Management, 5(1): 201-210. https://doi.org/10.46729/ijstm.v5i1.1028
[19] Anitha, K., Vijayakumar, R., Jeslin, J.G., Elangovan, K., Jagadeeswaran, M., Srinivasan, C. (2024). Marine propulsion health monitoring: Integrating neural networks and IoT sensor fusion in predictive maintenance. In 2024 2nd International Conference on Device Intelligence, Computing and Communication Technologies (DICCT), pp. 1-6. https://doi.org/10.1109/dicct61038.204.10533037
[20] Ong, K.S.H., Niyato, D., Yuen, C. (2020). Predictive Maintenance for edge-based sensor networks: A deep reinforcement learning approach. In 2020 IEEE 6th World Forum on Internet of Things (WF-IoT), pp. 1-6. https://doi.org/10.1109/wf-iot48130.2020.9221098
[21] El Ahmar, E., Rachini, A., Attar, H. (2024). Cybersecurity enhancement in IoT Wireless Sensor Networks using machine learning. WSEAS Transactions on Information Science and Applications, 21(1): 480-487. https://doi.org/10.37394/23209.2024. 21.43

NOMENCLATURE

Tp	Throughput (number of useful bits received per unit time)
u	System Utilization (ratio of service time to time between arrivals)
L	Historical Data Window (number of past observations used in prediction)
M	Prediction Steps (number of steps ahead predicted)
δ	Reliability Level Parameter
PL	Packet Payload
HL	Header Length
Ts	Transmission Time
$Tsrv$	Transmission Service Time
$RSSI$	Received Signal Strength Indicator
LQI	Link Quality Indicator
NF	Noise Floor