An Optimized Neuro Fuzzy System For Predictive Maintenance For Smart Manufacturing Systems

Krishna Bhayal[1], Prof.Manish Soni[2]
[1]Dept of Mech. Engg.
[2]Associate professor, Dept of Mech. Engg.
[1, 2] Mahakal Institute of Technology, Ujjain,M.P. (India)

Abstract- *Smart manufacturing systems represent the modern evolution of industrial automation where machines, sensors, communication technologies, and intelligent algorithms work together to improve productivity and operational efficiency. Industries are increasingly adopting predictive maintenance techniques to reduce machine downtime, improve reliability, and minimize maintenance costs. Traditional maintenance approaches such as corrective maintenance and preventive maintenance often fail to provide accurate predictions regarding machine failures. Corrective maintenance acts only after a failure occurs, while preventive maintenance follows fixed schedules that may lead to unnecessary servicing. To overcome these limitations, intelligent predictive maintenance systems based on data analytics and machine learning. This paper presents a hybrid neuro fuzzy inference systems (ANFIS) model for automated fault prediction for smart manufacturing systems which aim predictive maintenance. The proposed model improves upon the error performance of existing work in the domain.*

Keywords: Smart Manufatcuring, Predictive Maintenbance, Adaptive Neuro Fuzzy Inferene Systems, Regression, Root Mean Squared Error (RMSE).

I. INTRODUCTION

Data analytics and predictive maintenance are transforming smart manufacturing systems by enabling industries to shift from reactive and preventive strategies to intelligent, data-driven decision-making. In modern factories, large volumes of data are continuously generated through sensors, machines, and production lines. By leveraging advanced analytics, manufacturers can extract meaningful insights from this data to optimize operations, reduce downtime, and enhance productivity [1]. Predictive maintenance, in particular, uses historical and real-time data to anticipate equipment failures before they occur, ensuring smoother and more efficient manufacturing processes. This especially important for Industry 4.0 and Cyber Physical Systems (CPS) [2].

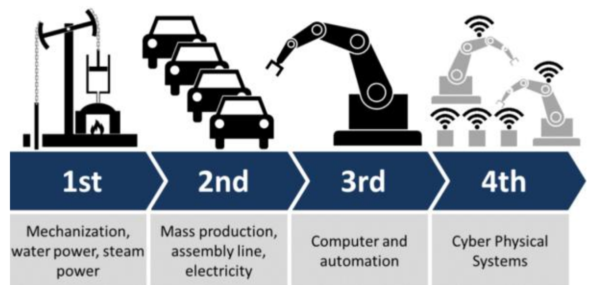

Fig.1 Concept of Industry 4.0
(Source: https://www.renaix.com/industry-4-0-the-fourth-industrial-revolution/)

Figure 1 depicts the concept of Industry 4.0.Industry 4.0 is where our physical and digital worlds combine – the typical Industry 4.0 standard bearers of progress, are the Internet of Things, Artificial Intelligence, Machine Learning and Big Data, and how these creations intersect with human networks and industrial capability worldwide. In short, Industry 4.0 optimizes what happened in Industry 3.0 by creating more interconnectivity and decentralization between cultures, knowledge centres, manufacturers and users [3].

Of course, the future of Industry 4.0 and the move to Industry 5.0 still needs to be discovered. The incredible changes that have come about in manufacturing in the 21st century are now firmly driven by digitization and efficiencies generated as a result. They impact everything, from manufacturing processes to principles and technologies, not to mention how manufacturing is powered, delivered, used or understood [4].

Data analytics plays a crucial role in smart manufacturing by processing and interpreting massive datasets generated from Industrial Internet of Things (IIoT) devices. Techniques such as descriptive, diagnostic, predictive, and prescriptive analytics allow manufacturers to monitor system performance, identify anomalies, and make informed decisions. With the integration of machine learning and artificial intelligence, analytics systems can uncover hidden patterns and correlations in data, enabling improved process optimization, quality control, and resource management. This data-driven approach significantly enhances operational efficiency and competitiveness [5].

II. PREDICTIVE MAINTENANCE AND FAULT IDENTIFICATION

Predictive maintenance refers to the use of data analysis tools and techniques to detect potential equipment failures before they happen. Unlike traditional maintenance strategies, which are either reactive (fix after failure) or preventive (scheduled maintenance), predictive maintenance relies on real-time monitoring and condition-based analysis. By analyzing parameters such as vibration, temperature, pressure, and acoustic signals, predictive models can estimate the remaining useful life (RUL) of machinery components. This helps in scheduling maintenance activities only when necessary, reducing unnecessary costs and downtime [6]

In smart manufacturing systems, predictive maintenance is integrated with cyber-physical systems, cloud computing, and edge computing platforms. This integration enables real-time data collection, processing, and analysis across the manufacturing ecosystem. Cloud platforms provide scalable storage and computational capabilities, while edge computing ensures low-latency processing near the data source. The seamless interaction between physical machines and digital systems enhances visibility, control, and automation in manufacturing processes, leading to improved reliability and efficiency [7].

Integration of Data Analytics and Predictive Maintenance:

The adoption of data analytics and predictive maintenance offers numerous benefits to manufacturing industries. It reduces unplanned downtime, minimizes maintenance costs, and extends the lifespan of equipment. Additionally, it improves product quality, enhances safety, and increases overall equipment effectiveness (OEE). By predicting failures in advance, manufacturers can avoid costly disruptions and maintain continuous production. Furthermore, it supports sustainable manufacturing by optimizing resource utilization and reducing waste [8].

The future of data analytics and predictive maintenance in smart manufacturing is promising, with advancements in artificial intelligence and big data technologies. Emerging trends such as federated learning, explainable AI, and autonomous maintenance systems are expected to further enhance predictive capabilities [9]. The use of digital twins and real-time simulation models will enable more accurate predictions and better decision-making. As industries continue to embrace Industry 4.0, the integration of intelligent analytics and predictive maintenance will become essential for achieving smart, adaptive, and resilient manufacturing systems [10]

Existing Challenges:

Despite its advantages, implementing data analytics and predictive maintenance in smart manufacturing systems presents several challenges. These include data quality issues, high initial investment costs, and the complexity of integrating heterogeneous systems. Moreover, the need for skilled professionals to develop and manage advanced analytics models can be a barrier for many organizations. Cybersecurity concerns and data privacy issues also pose significant risks, especially when dealing with interconnected systems and cloud-based platforms [11]

III. PROPOSED SYSTEM MODEL

The proposed system model is presented next:

A. Data Choice of Model:

Designing an optimized neuro-fuzzy system begins with data acquisition from industrial sensors monitoring parameters like temperature, vibration, pressure, and current. The collected data undergo preprocessing steps such as normalization, noise filtering, and feature extraction. These features are then fed into the neuro-fuzzy model, where fuzzy rules are generated and tuned using learning algorithms. The system is trained using historical data to recognize patterns associated with normal and faulty conditions [12].

A neuro-fuzzy system integrates artificial neural networks with fuzzy inference systems to create a hybrid model capable of both learning from data and handling ambiguity. Neural networks are proficient in identifying patterns and learning from large datasets, while fuzzy logic allows the system to represent human-like reasoning using linguistic variables such as “high temperature” or “low vibration.” The most commonly used architecture is the Adaptive Neuro-Fuzzy Inference System (ANFIS), which employs a layered structure to map inputs to outputs through fuzzy rules. This combination enhances interpretability and adaptability, making it suitable for industrial predictive maintenance applications [13]

B. Adaptive Neuro Fuzzy Inference Systems (ANFIS)

A very important tool that proves to be effective in several classification problems is fuzzy logic. It is often termed as expert view systems. It is useful for systems where there is no clear boundary among multiple variable groups. The relationship among the inputs and outputs are often expressed as membership functions expressed as [14]:

A membership function for a fuzzy set A on the universe of discourse (Input) X is defined as:

$$\mu A : X \rightarrow [0, 1] \quad (1)$$

Here,

each element of X is mapped to a value between 0 and 1. It quantifies the degree of membership of the element in X to the fuzzy set A [15].

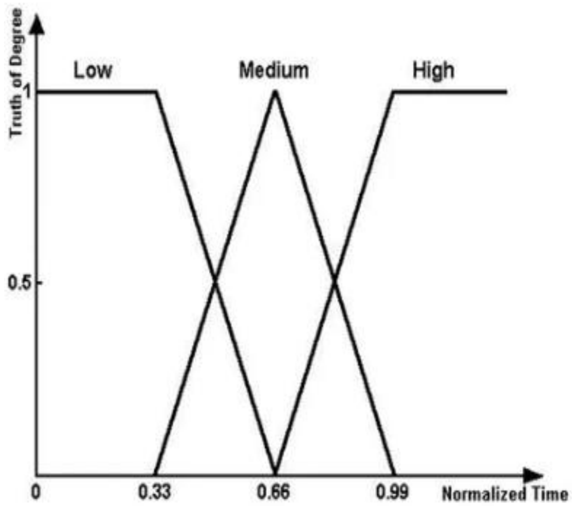

Fig.2 Graphical Representation of Membership Functions

Here,

x axis represents the universe of discourse (Input).

y axis represents the degrees of membership in the [0, 1] interval.

The final category is neuro fuzzy expert systems which governs the defining range of the membership functions [16].

While neuro-fuzzy systems are powerful, their performance heavily depends on the proper selection of parameters such as membership functions, rule sets, and network structure. Without optimization, these systems may suffer from issues like overfitting, increased computational complexity, and reduced prediction accuracy. Optimization techniques are therefore essential to fine-tune the system parameters, improve convergence speed, and enhance generalization capability [17].

The ANFIS can be thought of as a combination of neural networks and fuzzy logic. In this mechanism, the neural network module decides the membership functions of the fuzzy module. The ANFIS structure is depicted in figure 3.

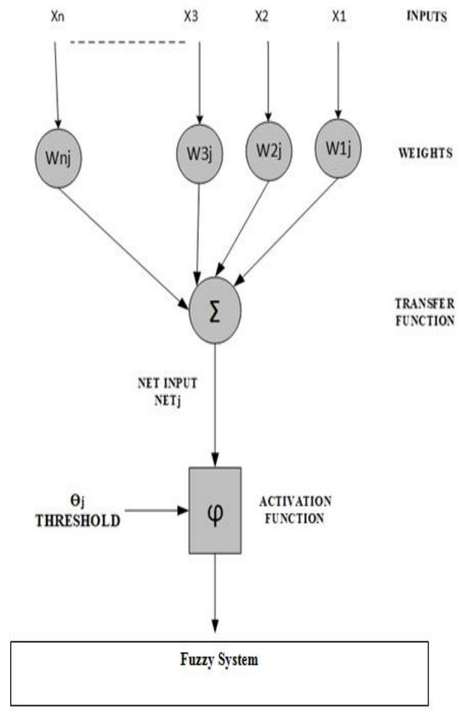

Fig.3 Block Diagram of Neuro-Fuzzy Expert Systems

The splitting can be done through the Gini's index which is especially useful for overlapping data sets since it can split data sets with overlapping classes based on conditional probability. The Gini's index for splitting is defined as [18]:

$$GI = 1 - \sum_{i=1}^{n} p_i^2 \quad (2)$$

Here,

GI represents the Gini's Index

P is the probability of a class

The prepared data vector for training is used for training wherein the weights are initialized randomly. A stepwise implementation is done as [19]:

1. Prepare two arrays, one is input and hidden unit and the second is output unit.

Here, a two dimensional array W_{ij} is used as the weigt updating vector andoutput is a one dimensional array Y_i.

3. Original weights are random values put inside the arrays after that the output [20].

$$x_j = \sum_{i=0} y_i W_{ij} \quad (3)$$

Where,

y_i is the activity level of the j^{th} unit in the previous layer and

W_{ij} is the weightof the connection between the i^{th} and the j^{th} unit.

4. Next, activation is invoked by the sigmoid function applied to the total weighted input [21].

$$y_i = \left[\frac{e^x - e^{-x}}{e^x + e^{-x}}\right] \quad (4)$$

Summing all the output units have been determined, the network calculates the error (E).

$$E = \frac{1}{2}\sum_i (y_i - d_i)^2 \quad (5)$$

Where, y_i is the event level of the j^{th} unit in the top layer and d_i is the preferred output of the j_i unit.

C. Implementing Back Prop:

Calculation of error for the back propagation algorithm is as follows:

Error Derivative (EA_j) is the modification among the real and desired target [22]:

$$EA_j = \frac{\partial E}{\partial y_j} = y_j - d_j \quad (6)$$

Here,
E represents the error
y represents the Target vector
d represents the predicted output

Error Variations is total input received by an output changed given by:

$$EI_j = \frac{\partial E}{\partial X_j} = \frac{\partial E}{\partial y_j} X \frac{dy_j}{dx_j} = EA_j y_j (1 - y_i) \quad (7)$$

Here,
E is the error vector
X is the input vector for training the neural network

In Error Fluctuations calculation connection into output unit is computed as [23]:

$$EW_{ij} = \frac{\partial E}{\partial W_{ij}} = \frac{\partial E}{\partial X_j} = \frac{\partial X_j}{\partial W_{ij}} = EI_j y_i \quad (8)$$

Here,
W represents the weights
I represents the Identity matrix
I and j represent the two dimensional weight vector indices

Overall Influence of the error:

$$EA_i = \frac{\partial E}{\partial y_i} = \sum_j \frac{\partial E}{\partial x_j} X \frac{\partial x_j}{\partial y_i} = \sum_j EI_j W_{ij} \quad (9)$$

The partial derivative of the Error with respect to the weight represents the error swing for the system while training. The gradient is computed as [24]:

$$g = \frac{\partial e}{\partial w} \quad (10)$$

Here,
g represents the gradient
e represents the error of each iteration
w represents the weights.

The training of a backpropagation neural network involves the iterative application of the backpropagation algorithm.

Considering the loss/cost function as the mean squared error, the weight update algorithm is given by [25]:

$$w_{k+1} = w_k - \left[J_k J_k^T + \mu I\right]^{-1} J_k^T e_k \quad (11)$$

Here,
k is the iteration number
w_{k+1} is weight of next iteration,
w_k is weight of present iteration
J_k is the Jacobian Matrix and is given by the terms $J_k = \frac{\partial^2 e}{\partial w^2}$ i.e. the second order rate of change of errors with respect to weights
J_k^T is Transpose of Jacobian Matrix
e_k is error of Present Iteration
μ is step size i.e. amount by which weight changes in each iteration
I is an identity matrix, with all diagonal elements equal to 1 and other elements 0.

During the training process, historical data is used to feed the network, and the algorithm calculates the error between the predicted and actual values. This error is then propagated backward through the network, adjusting the weights and biases of the connections to minimize the prediction error. This iterative process continues until the network converges to a state where the error is minimized.The training is stopped based on the mean square error or mse given by:

$$mse = \frac{\sum_{i=1}^{n} e_i^2}{n} \quad (12)$$

The proposed algorithm for the system is presented next:

Proposed Algorithm:

The algorithm of the proposed approach is presented subsequently:

Start

{

Step.1 *Extract dataset and divide data into the ratio of 70:30 for training : testing.*

Step.2 *Assign input and target variables.*

Step.3 *Initialize weight matrix randomly.*

Step.4 *To train the network, employ the following training rule:*

$$w_{k+1} = w_k - \left[J_k J_k^T + \mu I\right]^{-1} J_k^T e_k$$

Step.5*If (cost function stabilizes)*

Truncate training

Else if (max. iterations are over)

Truncate Training

Else

Feedback errors as inputs to subsequent iteration.

Step.6 *if (error is stable through validation checks i.e. consecutive iterations)*

Stop training

else if (maximum iterations are over even without error stabilization)

Stop Training

else

{

Feed next training vector

Back propagation of error

}

Step.7: *Simulate model to forecast samples.*

Step.8 *Compute performance metrics.*

}

Stop

The final computation of the performance metric is the mean absolute percentage error given by:

$$\boldsymbol{MAPE} = \frac{100}{M}\sum_{i=1}^{N}\frac{E-E_i|}{i} \quad \textbf{(13)}$$

The accuracy of prediction is computed as:

$$\boldsymbol{Ac} = 100 - \frac{100}{M}\sum_{i=1}^{N}\frac{E-E_i|}{i}\ \% \qquad \textbf{(14)}$$

Here,

n is the number of errors

i is the iteration number

E is the actual value

E_i is the predicted value

IV. RESULTS

The system is implemented on Matlab. The results obtained on implementing the proposed system is discussed in this section. The data parameters used in this study are:

1. Air temperature [K]
2. Process temperature [K]
3. Rotational speed [rpm]
4. Torque [Nm]: torque values are normally distributed around 40 Nm with an σ = 10 Nm and no negative values.
5. Tool Wear (High/ Medium/ Low).

The metric to be predicted is:

Failure or No-Failure.

	A	B	C	D	E	F	G	H	I	J
1	UDI	Product ID	Type	Air temper	Process te	Rotational	Torque [N	Tool wear	Target	Failure Type
2	1	M14860	M	298.1	308.6	1551	42.8	0	0	No Failure
3	2	L47181	L	298.2	308.7	1408	46.3	3	0	No Failure
4	3	L47182	L	298.1	308.5	1498	49.4	5	0	No Failure
5	4	L47183	L	298.2	308.6	1433	39.5	7	0	No Failure
6	5	L47184	L	298.2	308.7	1408	40	9	0	No Failure
7	6	M14865	M	298.1	308.6	1425	41.9	11	0	No Failure
8	7	L47186	L	298.1	308.6	1558	42.4	14	0	No Failure
9	8	L47187	L	298.1	308.6	1527	40.2	16	0	No Failure
10	9	M14868	M	298.3	308.7	1667	28.6	18	0	No Failure
11	10	M14869	M	298.5	309	1741	28	21	0	No Failure
12	11	H29424	H	298.4	308.9	1782	23.9	24	0	No Failure
13	12	H29425	H	298.6	309.1	1423	44.3	29	0	No Failure
14	13	M14872	M	298.6	309.1	1339	51.1	34	0	No Failure
15	14	M14873	M	298.6	309.2	1742	30	37	0	No Failure
16	15	L47194	L	298.6	309.2	2035	19.6	40	0	No Failure
17	16	L47195	L	298.6	309.2	1542	48.4	42	0	No Failure
18	17	M14876	M	298.6	309.2	1311	46.6	44	0	No Failure
19	18	M14877	M	298.7	309.2	1410	45.6	47	0	No Failure
20	19	H29432	H	298.8	309.2	1306	54.5	50	0	No Failure
21	20	M14879	M	298.9	309.3	1632	32.5	55	0	No Failure
22	21	H29434	H	298.9	309.3	1375	42.7	58	0	No Failure
23	22	L47201	L	298.8	309.3	1450	44.8	63	0	No Failure
24	23	M14882	M	298.9	309.3	1581	30.7	65	0	No Failure
25	24	L47203	L	299	309.4	1758	25.7	68	0	No Failure
26	25	M14884	M	299	309.4	1561	37.3	70	0	No Failure
27	26	L47205	L	299	309.5	1861	23.3	73	0	No Failure
28	27	L47206	L	299.1	309.5	1512	39	75	0	No Failure

data

Fig.4 Raw data samples

The raw data samples are collected after which it is imported to the Matlab workspace. The five independent variables are to be fed to the ANIS model to predict potential fault or non-fault of the system.

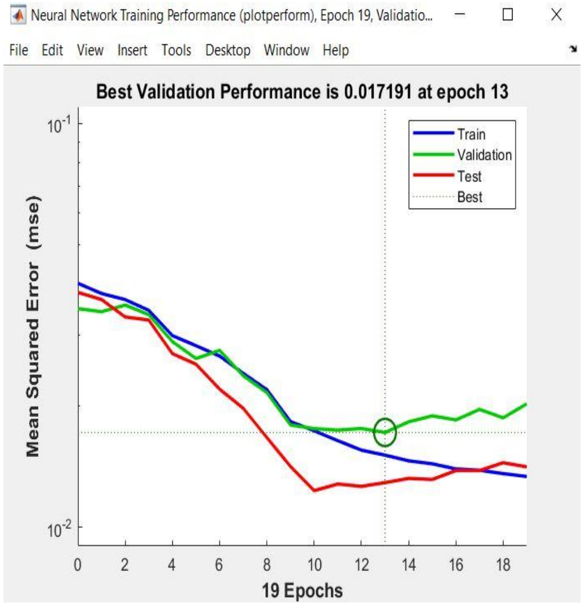

Fig.5Training Epochs

It can be observed that the model attains convergence in 19 iterations, with and MSE value of 0.017.

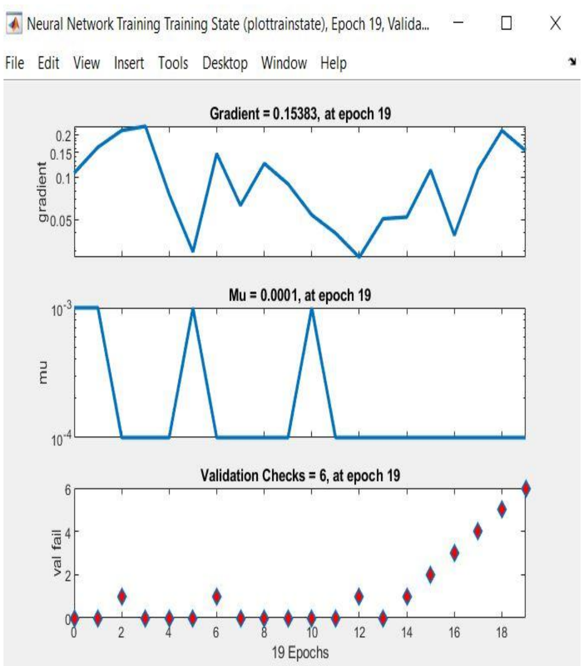

Fig.6 Training Parameters

The values of the gradient, learning rate and validation checks is depicted in the figure above.

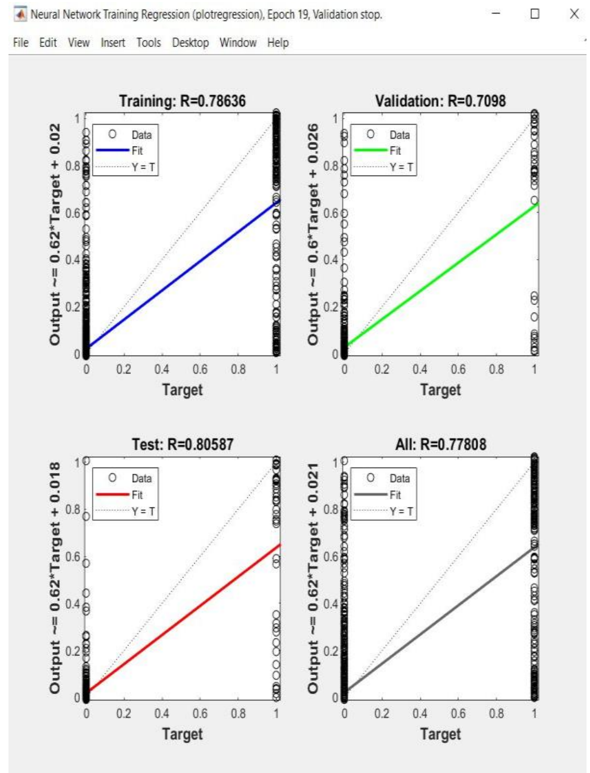

Fig.7Regression

Figure 7presents the regression obtained for the training / testing and validation processes. The mean overall regression value is 0.778.

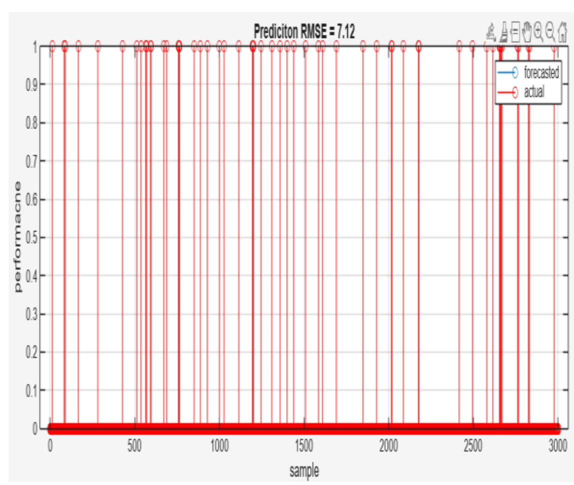

Fig.8Prediction RMSE

Figure 8 depicts the prediction RMSE value which is 7.12. .

A summary of results and comparison with existing work in the domain is presented next.

Table I: Summary of Results

S.No.	Parameter	Value
1.	Features	5
2.	Data samples	10,000
3.	Training: Testing Split	70:30
4.	Model	ANFIS
5.	Training	Back Prop
6.	**Iterations**	**19**
7.	Gradient	0.153
8.	Learning Rate	0.001
9.	**RMSE (Proposed System)**	**7.12**
10.	**RMSE of Previous Work (Jaenal et al.), [11]. MashNet (CNN-LSTM Hybrid)**	**11.04**

It can be observed that the proposed system attains improved prediction RMSE compared to existing work in the domain.

V. CONCLUSION

It can be concluded from previous discussions that predictive maintenance refers to the process of continuously monitoring machine conditions and predicting failures before they occur. In smart manufacturing systems, sensors installed on industrial equipment collect real-time operational data. This data is transmitted to centralized processing systems where machine learning and intelligent algorithms analyze equipment health conditions.The primary objective of predictive maintenance is to identify early signs of degradation and schedule maintenance activities at the optimal time. This approach significantly reduces unexpected equipment breakdowns and improves production continuity. Smart manufacturing environments generate massive amounts of heterogeneous data that are often noisy, uncertain, and nonlinear in nature. Therefore, intelligent data-driven models capable of handling uncertainty and complex relationships are essential. Neuro fuzzy systems are particularly suitable for such environments because they combine adaptive learning with approximate reasoning. The results of the proposed approach can be observed to be better compared to existing approaches in the domain.

REFERENCES

[1] A. Aboshosha, A. Haggag, N. George and H. A. Hamad, “IoT-Based Data-Driven Predictive Maintenance Relying on Fuzzy System and Artificial Neural Networks,” Scientific Reports, vol. 13, no. 1, pp. 1–18, Jul. 2023, doi: 10.1038/s41598-023-38887-z.

[2] M. Paolanti, L. Romeo, A. Felicetti, A. Mancini, E. Frontoni and J. Loncarski, “Machine Learning Approach for Predictive Maintenance in Industry 4.0,” in Proc. 14th IEEE/ASME Int. Conf. Mechatronic and Embedded Systems and Applications (MESA), 2018, pp. 1–6, doi: 10.1109/MESA.2018.8449150.

[3] W. Yu, T. Dillon, F. Mostafa, W. Rahayu and Y. Liu, “A Global Manufacturing Big Data Ecosystem for Fault Detection in Predictive Maintenance,” IEEE Transactions on Industrial Informatics, vol. 16, no. 1, pp. 183–192, Jan. 2020, doi: 10.1109/TII.2019.2915846.

[4] L. C. Him, Y. Y. Poh and L. W. Pheng, “IoT-Based Predictive Maintenance for Smart Manufacturing Systems,” in Proc. APSIPA Annual Summit and Conference (APSIPA ASC), 2019, pp. 1–6, doi: 10.1109/APSIPAASC47483.2019.9023106.

[5] P. Poor, J. Basl and D. Zenisek, “Predictive Maintenance 4.0 as Next Evolution Step in Industrial Maintenance Development,” in Proc. Int. Research Conf. Smart Computing and Systems Engineering (SCSE), 2019, pp. 245–253, doi: 10.23919/SCSE.2019.8842659.

[6] Q. Qiao, J. Wang, L. Ye and R. X. Gao, “Digital Twin for Machining Tool Condition Prediction,” Procedia CIRP, vol. 81, pp. 1388–1393, 2019, doi: 10.1016/j.procir.2019.04.049.

[7] B. C. Menezes, J. D. Kelly, A. G. Leal and G. C. Le Roux, “Predictive, Prescriptive and Detective Analytics for Smart Manufacturing in the Information Age,” IFAC-PapersOnLine, vol. 52, no. 1, pp. 568–573, 2019, doi: 10.1016/j.ifacol.2019.06.123.

[8] H. Zheng, A. R. Paiva and C. S. Gurciullo, “Advancing from Predictive Maintenance to Intelligent Maintenance with AI and IIoT,” arXiv preprint arXiv:2009.00351, 2020.

[9] Z. Zhai, B. Gehring and G. Reinhart, “Enabling Predictive Maintenance Integrated Production Scheduling by Operation-Specific Health Prognostics with Generative Deep Learning,” Journal of Manufacturing Systems, vol. 61, pp. 830–855, 2021, doi: 10.1016/j.jmsy.2021.02.006.

[10] L. Ferreira, A. Pilastri, F. Romano and P. Cortez, “Using Supervised and One-Class Automated Machine Learning for Predictive Maintenance,” Applied Soft Computing, vol. 131, p. 109820, 2022, doi: 10.1016/j.asoc.2022.109820.

[11] A. Jaenal, J.-R. Ruiz-Sarmiento and J. Gonzalez-Jimenez, “MachNet, a General Deep Learning Architecture for Predictive Maintenance Within the Industry 4.0 Paradigm,” Engineering Applications of Artificial Intelligence, vol. 127, p. 107365, 2024, doi: 10.1016/j.engappai.2023.107365.

[12] S. S. Dash, S. Dehuri and S.-B. Cho, “An Adaptive Neuro-Fuzzy Inference System for Predictive Data

Analytics: A Review," Applied Soft Computing, vol. 110, p. 107655, 2021, doi: 10.1016/j.asoc.2021.107655.

[13] M. R. Islam, M. M. Hasan and A. K. M. Fazlul Haque, "Data Analytics Using Adaptive Neuro-Fuzzy Inference System for Smart Healthcare Applications," IEEE Access, vol. 9, pp. 135019–135032, 2021, doi: 10.1109/ACCESS.2021.3116798.

[14] A. K. Sangaiah, G. Srivastava and Y. Xiang, "Intelligent Data Analytics Framework Using Neuro-Fuzzy Systems for Industrial Applications," Future Generation Computer Systems, vol. 123, pp. 64–75, 2021, doi: 10.1016/j.future.2021.04.018.

[15] H. M. Hasanien, S. H. E. Abdel Aleem and A. Y. Abdelaziz, "Adaptive Neuro-Fuzzy Inference System for Big Data Analytics in Smart Systems," Knowledge-Based Systems, vol. 235, p. 107645, 2022, doi: 10.1016/j.knosys.2021.107645.

[16] P. K. Donta, B. S. P. Rao and T. Amgoth, "Neuro-Fuzzy-Based Intelligent Data Analytics Model for IoT and Smart Manufacturing," Journal of Ambient Intelligence and Humanized Computing, vol. 13, no. 9, pp. 4501–4515, 2022, doi: 10.1007/s12652-021-03465-8.

[17] R. K. Kaliyar, A. Goswami and P. Narang, "An Explainable Neuro-Fuzzy Framework for Data Analytics and Decision Support Systems," Expert Systems with Applications, vol. 197, p. 116659, 2022, doi: 10.1016/j.eswa.2022.116659.

[18] M. A. Ferrag, L. Maglaras and H. Janicke, "Deep Neuro-Fuzzy Models for Intelligent Data Analytics in Cyber-Physical Systems," IEEE Transactions on Industrial Informatics, vol. 19, no. 3, pp. 2145–2155, Mar. 2023, doi: 10.1109/TII.2022.3154412.

[19] Y. Chen, Z. Lv and A. K. Singh, "Neuro-Fuzzy Learning Framework for Real-Time Big Data Analytics in Industry 4.0," Information Sciences, vol. 624, pp. 325–340, 2023, doi: 10.1016/j.ins.2022.12.061.

[20] V. Chang, A. K. Gupta and M. Ramachandran, "Artificial Intelligence and Neuro-Fuzzy Techniques for Predictive Data Analytics in Smart Environments," Sustainable Computing: Informatics and Systems, vol. 39, p. 100894, 2023, doi: 10.1016/j.suscom.2023.100894.

[21] T. R. Gadekallu, P. K. R. Maddikunta and S. Bhattacharya, "Hybrid Neuro-Fuzzy Intelligence for Advanced Data Analytics and Decision Making," Engineering Applications of Artificial Intelligence, vol. 128, p. 107541, 2024, doi: 10.1016/j.engappai.2023.107541.

[22] E. Jovicic, D. Primorac, M. Cupic and A. Jovic, "Publicly Available Datasets for Predictive Maintenance in the Energy Sector: A Review," IEEE Access, vol. 11, pp. 73505–73520, 2023, doi: 10.1109/ACCESS.2023.3295113.

[23] M. Sharma, M. Sharma, K. S. Yadav and S. Shukla, "Industry 4.0 Technologies for Smart Manufacturing: A Systematic Review of Machine Learning Methods for Predictive Maintenance," in Proc. Int. Conf. Smart Systems and Advanced Software (ICSSAS), 2023, pp. 397–403, doi: 10.1109/ICSSAS57918.2023.10331740.

[24] K. Kamat, P. Shah, V. Lad, P. Desai, Y. Vikani and D. Savani, "Data Acquisition Using IoT Sensors for Smart Manufacturing Domain," in Advances in Science, Technology & Innovation, 2021, pp. 393–400, doi: 10.1007/978-3-030-66218-9_46.

[25] D. S. Satwaliya, H. P. Thethi, A. Dhyani, G. R. Kiran, M. Al-Taee and M. B. Alazzam, "Predictive Maintenance Using Machine Learning: A Case Study in Manufacturing Management," in Proc. 3rd Int. Conf. Advance Computing and Innovative Technologies in Engineering (ICACITE), 2023, pp. 872–876.