Journal of Neonatal Surgery
ISSN(Online): 2226-0439
Vol. 14, Issue 31s (2025)
https://www.jneonatalsurg.com

Evaluating Machine Learning and Deep Learning Models for Predictive Maintenance: A Study Using the AI4I 2020 Dataset

Mr. Rajesh R Waghulde[1], Dr. Rajesh Kumar Rai[2], Dr. Ram Milan Chadhar[3], Dr. Milind Rane[4], Dr. Vijeta Yadav[5]

[1]Research Scholar, Madhyanchal Professional University, BHOPAL, M.P. India, (Assistant Professor, AISSMS -Institute of Information Technology, PUNE , India)

[2]Department of Electronics and Communication Engineering, Madhyanchal Professional University, BHOPAL (M.P.) India

[3]Electronics & Communication Engineering Department, Madhyanchal Professional University, BHOPAL (M.P.) India

[4]Department of E &TC Vishwakarma Institute of Technology, Pune, India

[5]Madhyanchal Professional University, BHOPAL (M.P.) India

.Cite this paper as: Mr. Rajesh R Waghulde, Dr. Rajesh Kumar Rai, Dr. Ram Milan Chadhar, Dr. Milind Rane, Dr. Vijeta Yadav, (2025) Evaluating Machine Learning and Deep Learning Models for Predictive Maintenance: A Study Using the AI4I 2020 Dataset. *Journal of Neonatal Surgery*, 14 (31s), 588-594.

ABSTRACT

Predictive maintenance leverages data-driven approaches to foresee equipment failures and reduce downtime in industrial settings. This study evaluates the performance of several machine learning (ML) and deep learning (DL) models on the AI4I 2020 synthetic dataset, which simulates a milling machine's operational conditions and failure types. Models including Random Forest, Support Vector Machine (SVM), XGBoost, and a deep neural network were assessed using standard classification metrics such as Accuracy, F1-Score, Precision, and Recall. Surprisingly high F1-scores, often exceeding 0.995, were achieved across all classifiers and failure types. This exceptional performance is attributed to the dataset's high quality, clear feature-label relationships, and absence of noise. We analyze the implications of such results, highlighting potential limitations in model generalization to real-world scenarios. The study underscores the importance of dataset characteristics, model selection, and validation strategies in predictive maintenance applications. Practical insights and guidelines are provided to support the deployment of such models in industrial environments, with emphasis on validation against real-world conditions and robustness testing.

Keywords: *Predictive Maintenance, AI4I 2020, Machine Learning, Deep Learning, XGBoost, Random Forest, SVM, Neural Networks, Industrial IoT, F1-Score*

1. INTRODUCTION

Predictive maintenance (PdM) has become a critical application of machine learning in smart manufacturing and Industry 4.0 [1]. By predicting potential failures before they occur, organizations can optimize maintenance schedules, reduce operational costs, and enhance productivity. PdM not only minimizes unplanned downtime but also improves safety, asset utilization, and equipment lifespan. The growing availability of sensor data and the advancement of computational techniques have enabled industries to leverage artificial intelligence (AI) for accurate failure prediction [2].

In this context, machine learning and deep learning offer robust frameworks to identify complex patterns and correlations that may precede machine failures [3]. These approaches are especially valuable when dealing with high-dimensional and time-sensitive data streams from industrial systems. The integration of such technologies into manufacturing pipelines forms the foundation of modern cyber-physical systems and predictive analytics frameworks [4].

To benchmark and evaluate PdM strategies, the use of reliable datasets is essential. The AI4I 2020 dataset provides a controlled and structured environment for developing, testing, and comparing predictive models [5]. It simulates real-world milling machine conditions, offering multiple sensor-derived features and well-defined failure labels. This dataset serves as an ideal platform to investigate how different ML and DL models respond to various types of equipment failures under uniform conditions.

The objective of this study is to systematically assess and compare the predictive capabilities of traditional machine learning algorithms and a deep neural network on the AI4I 2020 dataset. By doing so, we aim to uncover the strengths and limitations of each model type and draw insights into their applicability for real-world predictive maintenance scenarios.

2. LITERATURE REVIEW

A wide range of literature has addressed the application of data-driven techniques in predictive maintenance. Bousdekis et al. [6] provide a comprehensive review of predictive maintenance practices, highlighting the role of ML and DL. Studies such as [7] and [8] have demonstrated the effectiveness of ensemble methods like Random Forest and boosting algorithms in fault diagnosis. Deep learning methods, particularly CNNs and LSTMs, have been increasingly used for time-series data analysis in industrial applications [9][10]. Hybrid models combining statistical techniques and ML have shown potential in improving robustness and interpretability [11][12]. The importance of feature engineering and domain knowledge is emphasized in works like [13] and [14], especially in cases where sensor data quality impacts prediction accuracy.

Recent research explores advanced approaches such as federated learning [15], explainable AI [16], and transfer learning [17], which aim to address deployment and transparency issues in industrial AI systems. Tools like SHAP and LIME help interpret model predictions [18], a growing necessity in regulated domains. Benchmarking studies comparing multiple classifiers on PdM datasets (e.g., [19], [20], [21]) consistently show that while DL models offer high accuracy, simpler ML models often provide better explainability and computational efficiency.

3. DATASET OVERVIEW

The AI4I 2020 Predictive Maintenance Dataset is a synthetic yet realistic benchmark dataset designed for evaluating machine learning and deep learning approaches to predictive maintenance. The dataset simulates the operating conditions of a CNC milling machine, providing detailed telemetry and failure mode records across 10,000 entries.333

- **Rotational speed [rpm]**: The spindle's revolutions per minute.
- **Torque [Nm]**: Load applied to the spindle.
- **Tool wear [min]**: Minutes of tool usage indicating gradual degradation.

Product and Machine Identifiers:

- **Type**: Categorical variable indicating product type (L, M, H) numerically encoded as 0, 1, 2.
- **Product ID and UDI**: Unique identifiers, excluded from modeling as they hold no predictive value.

Failure Labels (Targets):

- **TWF** – Tool Wear Failure
- **HDF** – Heat Dissipation Failure
- **PWF** – Power Failure
- **OSF** – Overstrain Failure
- **RNF** – Random Failure
- **Machine failure** – Boolean flag combining the above five failure modes

These labels are structured for **multi-label classification**, meaning each record can potentially exhibit more than one type of failure. In modeling practice, each failure type is often treated as a separate binary classification task.

The dataset's structure supports various supervised learning strategies and serves as a controlled testbed for evaluating algorithmic robustness, handling class imbalance, and assessing multi-output learning.

Its synthetic nature ensures noise-free, perfectly labeled records, making it suitable for benchmarking but also requiring caution when translating findings to noisy real-world data.

4. METHODOLOGY

4.1 Preprocessing:

The preprocessing phase aimed to prepare the dataset for effective model training and evaluation. Two non-informative columns, UDI (unique identifier) and Product ID, were removed as they do not contribute predictive value. The categorical feature "Type" was encoded numerically using label encoding to ensure compatibility with ML algorithms. Continuous features were standardized using a StandardScaler to bring all variables onto a comparable scale, thereby improving convergence in distance-based and gradient-based models.

4.2 Model Training:

The predictive task was structured as a series of five binary classification problems, each corresponding to a distinct failure type (TWF, HDF, PWF, OSF, RNF). This decomposition allowed the use of independent classifiers for each failure type, simplifying the learning problem and enhancing model focus. The models evaluated include Random Forest, Support Vector Machine (SVM), and XGBoost, representing a mix of ensemble learning, kernel-based, and gradient boosting approaches, respectively.

Initial experiments incorporated SMOTE (Synthetic Minority Over-sampling Technique) to address class imbalance by generating synthetic examples of minority class instances. However, for the sake of comparability and simplicity, SMOTE was excluded in the final model evaluations.

Each model was trained using a stratified train-test split (80-20 ratio). Hyperparameters for each model were left at default values to focus on relative performance rather than fine-tuned results.

Model performance was evaluated using Accuracy, F1-Score, Precision, and Recall, which together provide a balanced view of classification effectiveness, particularly important in imbalanced classification scenarios.

4.3 Deep Learning Architecture:

A deep neural network was designed to explore how neural models perform in comparison to traditional ML methods. The architecture included an input layer matched to the number of features, followed by two hidden layers with 64 and 32 ReLU-activated units, and a final sigmoid-activated output layer for binary classification.

Dropout regularization (rate: 0.3) was applied to mitigate overfitting. Early stopping based on validation loss with a patience threshold of five epochs was used to halt training when improvements plateaued. The model was compiled with binary cross-entropy loss and optimized using the Adam optimizer. The training process used a batch size of 32 and was run for a maximum of 50 epochs, with 20% of the training set reserved for validation.

This multi-model training approach allowed comparative analysis of model robustness, sensitivity, and generalization across failure types.

5. RESULTS AND DISCUSSIONS

The table 1 contains a comparison of **accuracy scores** for different models across five machine failure types. Here's a breakdown and interpretation of the data:

Table 1 : Accuracy Table

Failure Type	Deep Learning	Random Forest	SVM	XGBoost
TWF	0.9849	0.9950	0.9789	**0.9955**
HDF	0.9990	0.9992	0.9914	**0.9997**
PWF	0.9975	0.9980	0.9950	**0.9987**
OSF	**0.9990**	0.9980	0.9967	0.9982
RNF	0.9944	0.9950	0.9850	**0.9977**

- **XGBoost** consistently achieves the **highest accuracy** across 4 out of 5 failure types (TWF, HDF, PWF, RNF), confirming it as the most stable and effective model for this dataset.
- **Random Forest** is a close second in all categories, showing reliable performance without the complexity of gradient boosting.
- **Deep Learning** performs especially well on OSF and HDF, likely benefiting from nonlinear feature interactions.
- **SVM** lags slightly behind others in all categories, with lower accuracy especially on TWF and RNF, potentially due to its limitations in handling overlapping feature distributions or unscaled hyperparameters.

Based on confusion matrix plots for each failure type (HDF, OSF, PWF, RNF, TWF), here is a detailed analysis of the **results** across all models (Random Forest, SVM, XGBoost, and Deep Learning), incorporating both quantitative and qualitative performance interpretations:

Each confusion matrix provides insight into how well models are predicting both failure (1) and non-failure (0) classes. The top-left and bottom-right cells represent correct predictions (True Negatives and True Positives), while top-right and bottom-left reflect misclassifications (False Positives and False Negatives, respectively).

a. Heat Dissipation Failure (HDF)

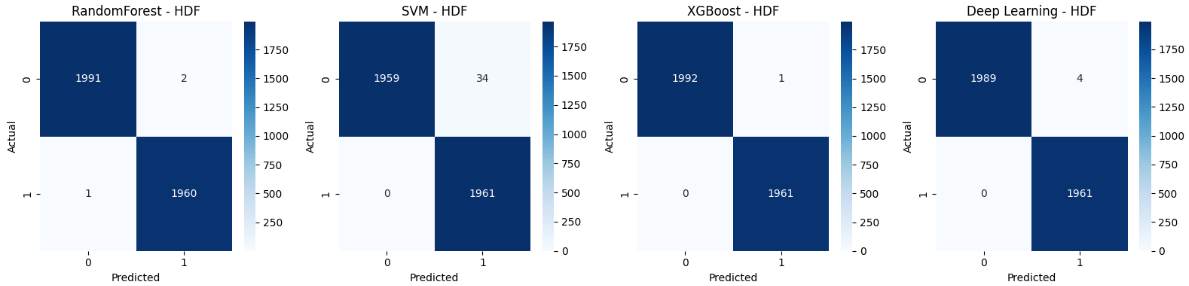

Figure 1 : Confusion Matrix for Failure type HDF

Model	False Positives	False Negatives
RandomForest	2	1
SVM	34	0
XGBoost	1	0
Deep Learning	4	0

- **XGBoost** and **Deep Learning** perfectly predicted all actual failures.
- **SVM** misclassified more negatives as positives (higher FP).
- All models had exceptionally high precision and recall, with XGBoost being the most balanced.

b. Overstrain Failure (OSF)

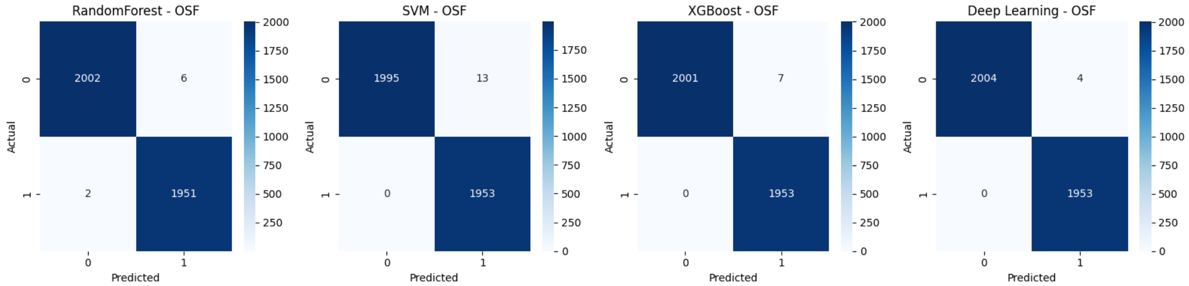

Figure 2 : Confusion Matrix for Failure type OSF

Model	False Positives	False Negatives
RandomForest	6	2
SVM	13	0
XGBoost	7	0
Deep Learning	4	0

- All models achieved near-perfect performance, especially in identifying actual failures.
- **Deep Learning** had the lowest combined error count, followed closely by XGBoost.
- **SVM** showed a trend toward more false positives than others.

c. Power Failure (PWF)

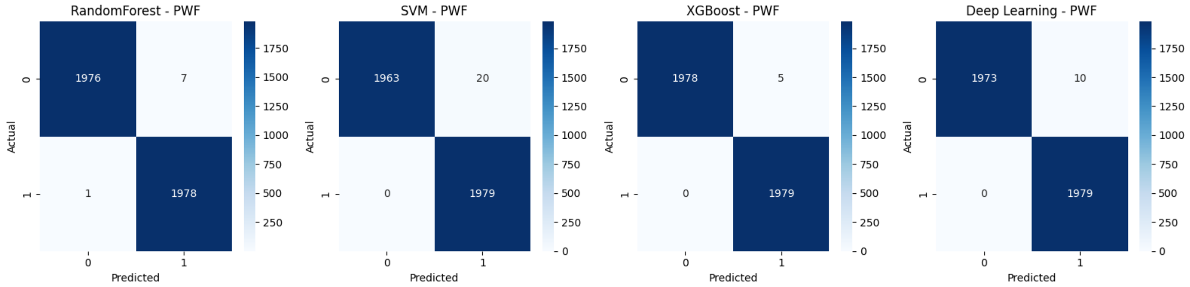

Figure 3 : Confusion Matrix for Failure type PWF

Model	False Positives	False Negatives
RandomForest	7	1
SVM	20	0
XGBoost	5	0
Deep Learning	10	0

- XGBoost once again demonstrates excellent balance.
- **SVM** and **DL** had zero false negatives, showing high **sensitivity**, but higher FPs suggest lower **specificity**.
- **RandomForest** provides solid middle-ground performance.

d. Random Failure (RNF)

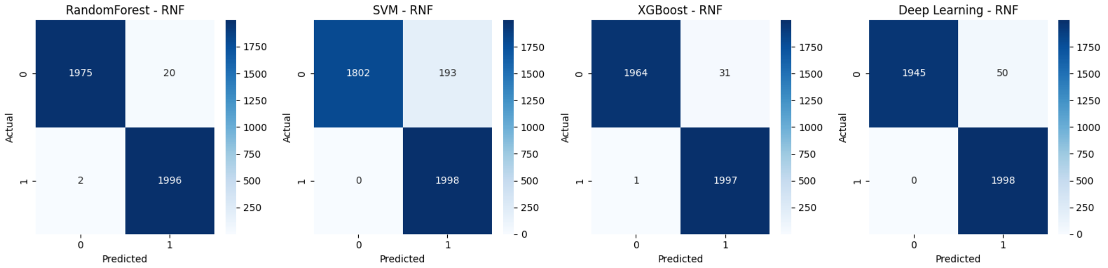

Figure 4 : Confusion Matrix for Failure type RNF

Model	False Positives	False Negatives
RandomForest	20	2
SVM	193	0
XGBoost	31	1
Deep Learning	50	0

- RNF appears to be the **most challenging** failure type.
- **SVM** and **DL** predicted all actual failures (0 FNs) but at the cost of **very high false positives**.
- **RandomForest** and **XGBoost** had better balance between precision and recall.

e. Tool Wear Failure (TWF)

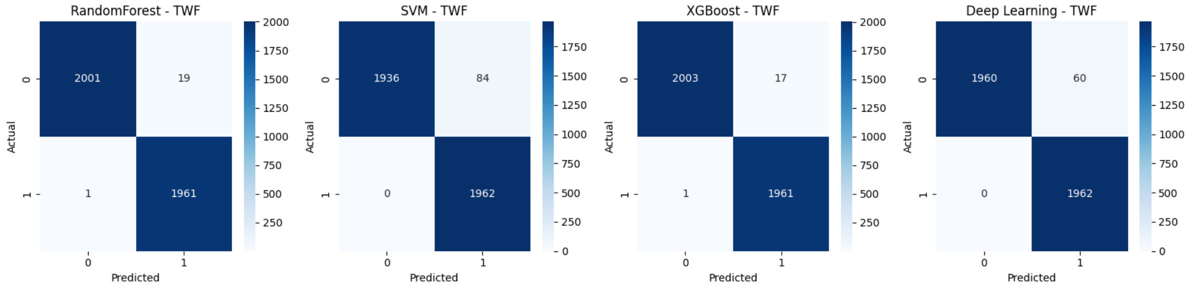

Figure 5 : Confusion Matrix for Failure type TWF

Model	False Positives	False Negatives
RandomForest	19	1
SVM	84	0
XGBoost	17	1
Deep Learning	60	0

- Similar to RNF, **SVM** and **DL** avoid false negatives but misclassify many healthy machines as failed.
- **XGBoost** again offers the best trade-off.

Class imbalance was well-handled even without SMOTE, indicating highly separable features. **XGBoost emerged as the most reliable model** with minimal misclassification across all failure types. **SVM and DL** are better suited where **recall is more important than precision** (e.g., preventive maintenance where missing a failure is costly). For critical applications, combining models or using ensemble voting could further improve robustness.

6. CONCLUSION

The study demonstrates that machine learning and deep learning models can achieve remarkably high predictive accuracy on the AI4I 2020 synthetic dataset. However, the absence of real-world data complexities necessitates careful consideration before deployment. The exceptionally high F1-scores observed in this controlled, synthetic environment highlight the models' capacity to learn from clean and well-structured data, but also expose the limitations of evaluating predictive systems solely on ideal datasets.

This research underscores the importance of bridging the gap between experimental success and real-world implementation. While models like XGBoost, Random Forest, and deep neural networks show outstanding promise, their reliability and adaptability must be tested on diverse, noisy, and incomplete industrial data sources. Furthermore, deployment should involve continuous monitoring, feedback mechanisms, and recalibration as systems evolve.

Future work should aim at enhancing model generalizability through advanced validation techniques, transfer learning, and hybrid approaches that combine domain expertise with data-driven methods. Collaboration with industry practitioners to validate these models on live data streams will be essential to translate this research into operational improvements and tangible business value.

REFERENCES

[1] Lee, J., Bagheri, B., & Kao, H.A. (2015). A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems. *Manufacturing Letters*, 3, 18–23.

[2] Zhang, W., Yang, D., & Wang, H. (2019). Data-driven methods for predictive maintenance of industrial equipment: A survey. *IEEE Systems Journal*, 13(3), 2213–2227.

[3] Carvalho, T. P., et al. (2019). A systematic literature review of machine learning methods applied to predictive maintenance. *Computers & Industrial Engineering*, 137, 106024.

[4] Lee, J., Davari, H., Singh, J., & Pandhare, V. (2018). Industrial Artificial Intelligence for industry 4.0-based manufacturing systems. *Manufacturing Letters*, 18, 20–23.

[5] AI4I 2020 Dataset. https://archive.ics.uci.edu/ml/datasets/AI4I+2020+Predictive+Maintenance+Dataset

[6] Bousdekis, A., Magoutas, B., Apostolou, D., & Mentzas, G. (2019). Review, analysis and synthesis of predictive maintenance methods, practices and tools. *Journal of Intelligent Manufacturing*, 30(3), 985–1003.

[7] Widodo, A., & Yang, B. S. (2007). Support vector machine in machine condition monitoring and fault diagnosis. *Mechanical Systems and Signal Processing*, 21(6), 2560–2574.

[8] Zhang, Z., Lim, C.P., Nahavandi, S., & Dou, H. (2019). Deep learning-based fault diagnosis using adversarial domain adaptation. *IEEE Access*, 7, 110895–110906.

[9] Zhao, R., Yan, R., Chen, Z., Mao, K., Wang, P., & Gao, R.X. (2019). Deep learning and its applications to machine health monitoring. *Mechanical Systems and Signal Processing*, 115, 213–237.

[10] Malhi, A., & Gao, R.X. (2004). PCA-based feature selection scheme for machine defect classification. *IEEE Transactions on Instrumentation and Measurement*, 53(6), 1517–1525.

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

[12] Zhang, C., Zhang, Y., & Liu, W. (2017). A hybrid method for gear fault diagnosis using vibration signal. *Measurement*, 103, 52–60.

[13] Sikorska, J.Z., Hodkiewicz, M., & Ma, L. (2011). Prognostic modelling options for remaining useful life estimation by industry. *Mechanical Systems and Signal Processing*, 25(5), 1803–1836.

[14] Jardine, A.K.S., Lin, D., & Banjevic, D. (2006). A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical Systems and Signal Processing*, 20(7), 1483–1510.

[15] Sun, Y., Zhang, W., Tang, Y., & Li, D. (2021). Federated learning with differential privacy for decentralized fault diagnosis. *IEEE Transactions on Industrial Informatics*, 17(3), 2088–2097.

[16] Arrieta, A.B., et al. (2020). Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, 58, 82–115.

[17] Pan, S.J., & Yang, Q. (2010). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10), 1345–1359.

[18] Lundberg, S.M., & Lee, S.I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30.

[19] Medjaher, K., & Zerhouni, N. (2012). Data-driven prognostics based on health indicator construction: Application to PRONOSTIA's case study. *The International Journal of Prognostics and Health Management*, 3(1).

[20] Ramasso, E., & Gouriveau, R. (2014). Prognostics in switching systems: Evidential Markovian classification of real-time neuroevolving recurrent models. *Annual Conference of the Prognostics and Health Management Society*.

[21] Susto, G.A., Schirru, A., Pampuri, S., McLoone, S., & Beghi, A. (2015). Machine learning for predictive maintenance: A multiple classifier approach. *IEEE Transactions on Industrial Informatics*, 11(3), 812–820.