Clareus Scientific Science and Engineering
Volume 2 Issue 5 May 2025
ISSN: 3065-1182

Predictive Algorithms for Maintenance Planning and Optimization in Industrial Applications

Citation: Alessandro Del Prete., et al. "Predictive Algorithms for Maintenance Planning and Optimization in Industrial Applications". Clareus Scientific Science and Engineering 2.5 (2025): 41-51.

Article Type: Research Article
Received: April 23, 2025
Published: May 29, 2025

Alessandro Del Prete*, Egidia Cirillo, Zahida Mashaallah and Alberto Moccardi

Department of Electrical Engineering and Information Technology (DIETI),University of Naples Federico II, via Claudio 21, Naples, 80125, Italy

***Corresponding Author:** Alessandro Del Prete, Department of Electrical Engineering and Information Technology (DIETI),University of Naples Federico II, via Claudio 21, Naples, 80125, Italy.

Abstract

Predictive maintenance PdM has helped, in recent decades, manufacturing and industry to save costs and keep their operations safe. This study outlines how advanced machine learning systems, including LSTMs and Transformers, could enhance data-driven maintenance planning. Mainly using C-MAPSS datasets to test Deep Learning (DL) methods and estimate Remaining Useful Life (RUL) values, this research aims to compare LSTM networks and Transformer performance in prognostic via different evaluation criteria. The analysis manifests the peculiarities of both the proposed learning approaches with a marked difference in the performances in favor of the recurrent architecture (e.g., 40% in term R^2 and 43% in terms of MSE), thus not generally suggesting the usage of transformer-based architecture, especially in a data-scarcity situation common condition when working of critical and costly units, but opening to a new perspective in otherwise operational conditions where data is prosperous.

Introduction

The aerospace industry moves forward with new technology while following strict safety rules and working to run operations better. For decades, traditional maintenance plans like preventive and reactive methods have driven industry operations [1]. However, these standard approaches cannot keep up with rising customer needs and complex production settings [16]: they fail to detect problems ahead of time, degrading the performance of planning [18] and wasting resources while increasing costs and risking worker safety. Artificial intelligence tools and big data analytics [17] are transforming how companies manage equipment maintenance [2], enabling predictive maintenance (PdM) systems that make data-based, proactive decisions. By analyzing real-time sensor streams, operational logs, and historical records, PdM forecasts failures before they occur with high accuracy [3].

The key feature of predictive maintenance, the estimation of the remaining useful life (RUL), allows a change from fixed schedules to condition-based maintenance, reducing downtime and expenses. Beyond classic recurrent networks, recent deep architectures have shown superior capabilities in modeling long-range dependencies and handling scarce or noisy data. Notable examples include Temporal Convolutional Networks (TCN) [19], Informer [21], Autoformer [22], N-BEATS [23], NHiTS [24], DeepAR [25], and the Temporal Fusion Transformer (TFT) [26]. These models combine convolutional, attention, and residual mechanisms to improve forecasting accuracy, interpretability, and computational efficiency under diverse industrial scenarios.

This study focuses on comparing Long Short-Term Memory and Transformer architectures for RUL prediction on the C-MAPSS dataset, refining maintenance schedules and addressing multifactorial conditions to develop more efficient and reliable aerospace maintenance strategies.

Related works

Predictive maintenance has emerged in recent years as one of the main investigation fields in aerospace, where one of the essential tasks is the precise forecast of the component RUL due to its safety and cost considerations for minimizing downtime [4]. More traditionally, basic methods like regression analysis and some physical models built the basis. However, with deep learning, the main focus has shifted to using more mature and data-driven methods which can surmount problems presented by modern aerospace systems.

One of the most widely explored models in this domain is the long-short-term memory (LSTM) network, a specialized type of recurrent neural network known for its ability to capture long-term dependencies in time-series data. Numerous studies [5-8] have demonstrated the effectiveness of LSTM models in predicting degradation patterns in critical aerospace components, including engines, bearings and rotors. Their ability to model temporal dynamics makes them especially well-equipped for tackling the sophisticated and dynamic operational conditions in the aerospace industry.

More recent Transformer models were developed for use in natural language processing tasks [9] but lately have gained immense attention in the field of time-series forecasting [10]. Transformers achieve this through their self-attention mechanisms, capturing intricate relationships even between data points separated by great time intervals. As such, with the ability to capture long-range dependencies, these Transformer models hold a powerful place in RUL prediction in aerospace applications, holding opportunities to continue advancing predictive maintenance strategies.

Advances in machine learning have introduced powerful architectures capable of modeling complex temporal dependencies:

- ***Temporal Convolutional Networks (TCN)*** utilize dilated convolutions to capture long-range patterns efficiently, outperforming RNNs on several benchmarks [19].
- ***Graph Neural Networks***, which exploit system topology to capture inter-sensor dependencies and improve system-level health indices [20].
- ***Informer*** employs ProbSparse self-attention to reduce complexity to $O(L \log L)$, enabling the processing of very long sequences for industrial time series [21].
- ***Autoformer*** integrates series decomposition and auto-correlation modules, improving long-horizon forecasting accuracy while maintaining interpretability [22].
- ***N-BEATS*** introduces a deep residual architecture with basis expansion blocks, achieving state-of-the-art in univariate forecasting competitions without domain-specific priors [23].
- ***N-HiTS*** extends N-BEATS with hierarchical interpolation and multi-rate sampling, reducing computation time by an order of magnitude for long-horizon tasks [24].
- ***DeepAR*** formulates probabilistic forecasting via autoregressive RNNs trained on large collections of related series, offering uncertainty estimates crucial for risk-aware maintenance planning [25].
- ***Temporal Fusion Transformer (TFT)*** combines recurrent layers and attention for multi-horizon forecasting, providing interpretable insights through gating and variable selection mechanisms [26].

These architectures, by integrating self-attention, convolutional, and residual mechanisms, surpass traditional LSTM-based approaches under diverse data regimes.

C-MAPSS literature review

The Commercial Modular Aero-Propulsion System Simulation (C-MAPSS) dataset, developed by NASA, is a fundamental resource for predictive maintenance research in the aerospace industry included in almost the 30% of the predictive maintenance papers. This dataset simulates the progressive performance degradation of turbofan engines in various operating conditions and fault scenarios, providing a comprehensive platform to develop and validate predictive algorithms [11].

C-MAPSS is divided into multiple subsets, each representing specific operational contexts and varying levels of fault severity. These subsets commonly include detailed time series data of 21 sensor readings, operational settings and RUL ground truth for test set (see Tab. 1).

Dataset	***FD001***	***FD002***	***FD003***	***FD004***
Train trajectories	100	260	100	249
Test trajectories	100	259	100	248
Sensor measurements	21	21	21	21
Maximum cycle	362	378	525	543
Minimum cycle	128	128	145	128
Mean cycle	206	206	247	245
Fault conditions	1	1	2	2

Table 1: Characteristics of C-MAPSS dataset.

This is particularly useful for modeling long-term dependencies in engine degradation, as the dataset has a time series nature, and is therefore well-suited for LSTM networks. Additionally, multisensor data can be used by Transformer models to uncover intricate interdependencies among parameters. The existence of various operating conditions and fault scenarios allows for a comprehensive assessment of the adaptability of the model, while its widespread use facilitates standardized performance comparisons between different predictive maintenance approaches [12].

Cross-domain Applications

Beyond aerospace, PdM models have been successfully applied to:

- ***Automotive***: RUL estimation for electric vehicle batteries and transmissions using sequence models and probabilistic forecasting [27, 28].
- ***Energy***: health monitoring of wind turbines and power grid assets employing TCN and TFT frameworks for long-term load forecasting [29].
- ***Manufacturing***: tool wear prediction in CNC machining and fault diagnosis in industrial robots via CNN ensembles and graph-based deep learning [30].
- ***Healthcare***: forecasting vital signs and sepsis risks in ICUs with Transformer model for interpretable health monitoring [31].
- ***Transportation***: traffic flow prediction using federated learning with Transformers and differential privacy to preserve user data confidentiality [32, 33].

These cross-domain successes highlight the versatility of advanced ML architectures for PdM across sectors.

Methodology

This paper presents the performance of two advanced deep learning architectures, namely long-short-term memory networks and transformer models, in predicting the remaining useful life of aerospace components.

The LSTM model was designed with two recurrent layers to effectively capture temporal dependencies in the input time series. To mitigate overfitting, two dropout layers were incorporated between the recurrent layers. A dense output layer with a linear activation function was used to provide the final prediction of RUL.

For the Transformer model, the architecture included a Transformer block featuring a multi-head attention mechanism and a feed-forward neural network. The attention mechanism allowed the model to focus on critical segments of the input sequence, efficiently capturing long-range dependencies. The feed-forward network comprised two 1D convolutional layers with ReLU activation, supported by normalization and dropout layers to enhance learning stability and prevent overfitting. Global average pooling was applied for dimensionality reduction, followed by fully connected layers to learn complex relationships within the data. This pipeline ended with the RUL prediction.

Overview of LSTM and Transformer Architectures

Understanding the mechanism and capability of LSTM networks and Transformer architectures requires knowledge of the underlying mathematical principles that are used in their design. In this section, the basic equations and operations that describe these architectures are given.

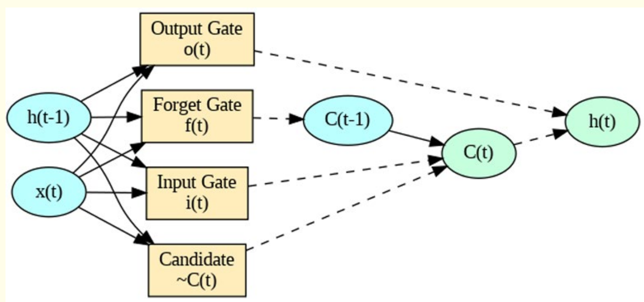

Figure 1: Block diagram of LSTM architecture.

An LSTM cell comprises three primary gates—Forget Gate, Input Gate, and Output Gate (Fig. 1)—each responsible for regulating the flow of information. The gates operate using the following equations:

$$f_t = \sigma\,(W_f \cdot [h_{t-1}, x_t] + b_f) \qquad (1)$$

$$i_t = \sigma\,(W_i \cdot [h_{t-1}, x_t] + b_i) \qquad (2)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \qquad (3)$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \qquad (4)$$

$$o_t = \sigma\,(W_o \cdot [h_{t-1}, x_t] + b_o) \qquad (5)$$

$$h_t = o_t \odot \tanh(C_t) \qquad (6)$$

with:

- x_t represents the input at the time step t.
- h_{t-1} is the hidden state from the previous time step.
- f_t, i_t, and o_t correspond to the forget, input, and output gate activations.
- $\tilde{C}_t$ is the proposed cell state, while C_t is the updated cell state.
- W_f, W_i, W_C, and Wo are the weight matrices for each gate, and b_f, b_i, b_C, and b_o are their respective bias terms.

Transformer

The Transformer architecture operates on the principles of self-attention and feedforward networks (Fig. 2), enabling it to model long-range dependencies efficiently.

The self-attention mechanism computes the attention scores for each token in a sequence. Given input embeddings $X = x_1, x_2, ..., x_n$, attention scores are derived as follows:

$$A_{ij} = \text{softmax}\left(\frac{Q_i K_j^T}{\sqrt{d_k}}\right) \quad (7)$$

where Q, K, and V are, respectively, query, key, value matrices, and d_k is the dimensionality of the key vectors.

To enhance interpretability, the attention mechanism is applied in parallel across multiple “heads” Each head computes attention with distinct input matrices, and the outputs are concatenated then linearly transformed.

The FFN, applied independently to each position, transforms the multi-head attention output using:

$$\text{FFN}(x) = \text{ReLU}(xW_1 + b_1)W_2 + b_2 \quad (8)$$

where W_1, W_2 are weight matrices, and b_1, b_2 are biases.

Experiments and Performance Evaluation

In this work, the four primary performance metrics are calculated, included in Table 2, for a fair LSTM and Transformer models’ performance evaluation, specifically Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE) and coefficient of determination (R^2), represents crucial KPIs for assessing the accuracy of the predictions and the goodness of fit to the C-MAPSS dataset for one-step RUL forecasting.

Mean Absolute Error

The mean absolute error (MAE) is a commonly used metric to evaluate regression models [13]. Measures the average magnitude of errors between the predicted and actual values, disregarding their direction. MAE is useful for understanding how close, on average, the predictions are to the true values. The formula is:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \quad (9)$$

where: n is the number of data points, y_i is the actual value and $\hat{y}_i$ is the predicted value. Generally a lower MAE indicates that the model produces more accurate predictions.

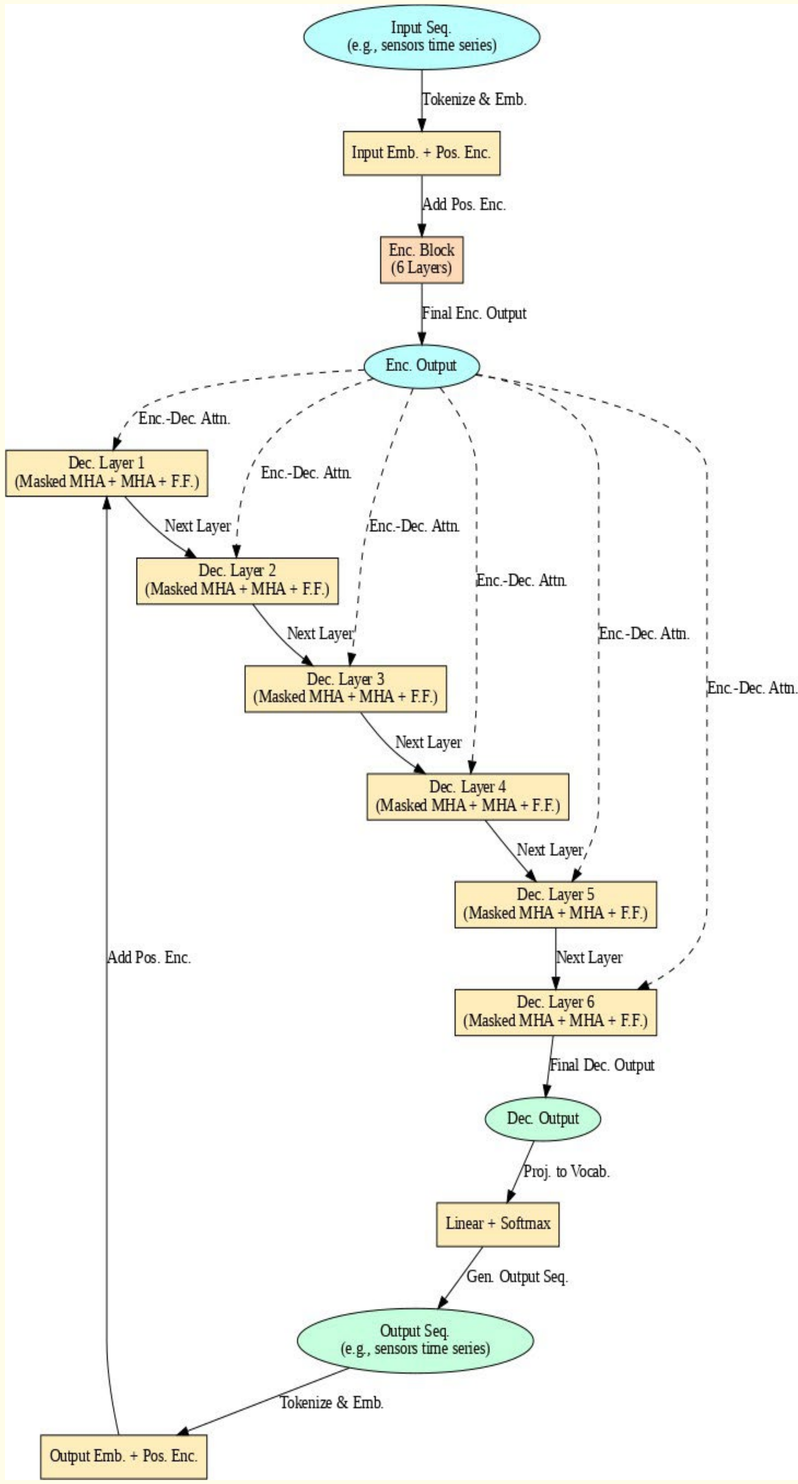

Figure 2: Overview of Transformer architecture.

Metric	*Definition*	*Formula*	*Usage*
Mean Absolute Error (MAE)	Average magnitude of errors in a set of predictions, without considering their direction.	$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n} \lvert y_i - \hat{y}_i \rvert$	Useful for understanding the average error in predictions.
Mean Squared Error (MSE)	Measures the average of absolute errors, giving higher weight to larger errors.	$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n} (y_i - \hat{y}_i)^2$	Penalizes large errors more than MAE; helpful for tasks where outliers are critical.
Root Mean Squared Error (RMSE)	The square root of MSE, keeping the error measure in the same units as the target variable.	$\text{RMSE} = \sqrt{\text{MSE}}$	Easier interpretation compared to MSE, especially when comparing with original data scale.
R^2	Proportion of variance in the dependent variable predictable from the independent one.	$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$	Indicates goodness of fit; values closer to 1 imply better performance.

Table 2: Overview of Common Evaluation Metrics.

Criterion	*Description*	*Reference*
Early stopping	Halt when validation loss has not improved for p consecutive epochs	[34]
Temporal cross-validation	Split time series into k temporal folds and perform early stopping per fold	[35]
ReduceLROnPlateau	Decrease learning rate on validation plateau; stop at min imum LR-threshold	[36]
Hyperband	Adaptive resource allocation via successive halving of tri als	[37]
Bayesian optimization	Treat stopping epoch as a hyperparameter modeled by a Gaussian process	[38]
Population Based Training	Evolve hyperparameters and stopping criteria across a population of models	[39]
Stochastic Weight Averaging	Average weights over epochs; stop when weight-variance falls below a threshold	[40]
Cyclical learning rates	Adjust LR cyclically; stop when performance metrics plateau within a cycle	[41]
Dynamic LR tuning	Automated LR trajectory algorithm; halt once conver gence behavior is detected	[42]
Performance threshold	Stop oncevalidation metric meets or exceeds a predefined target	[43]

Table 3: Alternative convergence criteria for comparing LSTM and Transformer models beyond fixed epochs.

Mean Squared Error

The mean squared error (MSE) penalizes larger errors more heavily than MAE by squaring the differences between the predicted and actual values [14].

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

While MSE is primarily a loss function, it provides an indirect measure of the model's performance. A lower MSE (or loss) indicates that the model has effectively learned patterns from the training data.

Root Mean Squared Error

The Root Mean Squared Error (RMSE) is the square root of the MSE and provides an interpretable measure of the model's prediction error in the same units as the target variable:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{11}$$

RMSE is especially useful when large errors are particularly undesirable, as it penalizes them more strongly.

Coefficient of Determination:

The coefficient of determination (R^2) measures how well the model explains the variance in the target variable. A R^2 score close to 1 suggests the model effectively captures the variance in the data, whereas a negative score indicates poor performance [15].

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{12}$$

The LSTM network and the Transformer model's performance were checked on the C-MAPSS dataset using metrics described in the previous section. The results are summarized in Table 4.

	MAE	***MSE***	***RMSE***	***R^2***
LSTM	20.5	922.8	30.36	0.73
Transformer	29.1	1609.3	40.11	0.52

Table 4: Comparison of Metric Results for LSTM and Transformer Models.

Interestingly, the loss curves and final evaluation metrics for both models show a very interesting discrepancy. The test loss for the Transformer is much lower compared to that of the LSTM model, as shown by (Fig. 3). This means that in terms of the loss minimization over training and testing, the Transformer does very well. The final metrics tell another story, however. The LSTM is superior to the Transformer with respect to lower errors and a higher (R^2) score, indicating better predictiveness on the evaluation dataset. This discrepancy can be attributed to several factors. First, it is possible that the Transformer, with its higher complexity, overfits the training data. While it excels in reducing the test loss, it fails to generalize as effectively as the simpler LSTM model when evaluated on unseen data. The LSTM, despite a slower reduction in loss and a higher test loss, demonstrates greater robustness and stability, leading to better performance on evaluation metrics.

Conclusions

Finally, the presented work concerns the application of long-short-term memory networks and transformer models for RUL prediction of turbofan engines using the C-MAPSS dataset. Our results show that, while both models have unique strengths, the LSTM outperforms the Transformer across all evaluation metrics, probably due to its ability to generalize more effectively on smaller, noisier datasets.

Figure 3: Comparison of test set loss between Transformer and LSTM models over 35 epochs.

Moreover, beyond fixed-epoch training, a variety of convergence tests, such as early stopping, learning-rate schedulers, and hyperband, offer practical alternatives to benchmark LSTM and Transformer performance (see Table 3).

The results further establish the importance of the architecture of the matching model and the characteristics of the data set. While the results are promising with the Transformer model, especially with larger and more diverse datasets, further optimization by the model together with access to these large and diverse datasets are needed.

Beyond aerospace, the proposed framework offers a versatile foundation for predictive maintenance in other critical sectors: electric vehicle battery and transmission health in automotive applications [27], monitoring of wind turbine condition in the energy industry [29], prediction of the wear of CNC tools in manufacturing [30], just to mention a few.

In future work, further exploration of these dynamics will be conducted addressing challenges such as data scarcity, computational complexity, and model interpretability. To enhance both LSTM and Transformer models for real-world predictive maintenance applications, efforts should focus on optimizing training processes, developing hybrid approaches, and leveraging synthetic or augmented datasets. Additionally, improving the efficiency of Transformer architectures and tailoring them for smaller datasets could bridge the performance gap observed in this study, unlocking their potential for aerospace and other industries.

Acknowledgements

This work was partially supported by PNRR MUR Project PE0000013FAIR. The FAIR project is committed to promoting an advanced vision of Artificial Intelligence, driving research and development in this crucial field and constantly keeping ethical, legal and sustainability considerations in mind.

References

1. Pop GI, Titu AM and Pop AB. "Enhancing Aerospace Industry Efficiency and Sustainability: Process Integration and Quality Management in the Context of Industry 4.0". Sustainability 15.23 (2023): 16206.
2. Ucar A, Karakose M and Kirimca N. "Artificial intelligence for predictive maintenance applications: key components, trustworthiness, and future trends". Applied Sciences 14.2 (2024): 898.

3. Meddaoui A, Hain M and Hachmoud A. "The benefits of predictive maintenance in manufacturing excellence: a case study to establish reliable methods for predicting failures". The International Journal of Advanced Manufacturing Technology 128.7-8 (2023): 3685-3690.
4. Lee WJ., et al. "Predictive maintenance of machine tool systems using artificial intelligence techniques applied to machine condition data". Procedia Cirp 80 (2019): 506-511.
5. Zhang J., et al. "Long short-term memory for machine remaining life prediction". Journal of manufacturing systems 48 (2018): 78-86.
6. ElDali M and Kumar KD. "Fault diagnosis and prognosis of aerospace systems using growing recurrent neural networks and LSTM". In 2021 IEEE aerospace conference. IEEE 50100 (2021): 1-20.
7. Wu Y., et al. "Remaining useful life estimation of engineered systems using vanilla LSTM neural networks". Neurocomputing 275 (2018): 167-179.
8. Dong D, Li XY and Sun FQ. "Life prediction of jet engines based on LSTM-recurrent neural networks". In 2017 prognostics and system health management conference (PHM-Harbin). IEEE (2017): 1-6.
9. Vaswani A., et al. "Attention is all you need. Advances in neural information processing systems". Advances in neural information processing systems 30 (2017).
10. Zeng A., et al. "Are transformers effective for time series forecasting?". In Proceedings of the AAAI conference on artificial intelligence 37.9 (2023): 11121-11128.
11. Ramasso E and Saxena A. "Performance Benchmarking and Analysis of Prognostic Methods for CMAPSS Datasets". International Journal of Prognostics and Health Management 5.2 (2014): 1-15.
12. Ferreira C and Goncalves G. "Remaining Useful Life prediction and challenges: A literature review on the use of Machine Learning Methods". Journal of Manufacturing Systems 63 (2022): 550-562.
13. James G., et al. An introduction to statistical learning 112 (2013): 18.
14. Hastie T., et al. "The elements of statistical learning: data mining, inference, and prediction". New York: Springer 2 (2009): 1-758.
15. Draper NR. "Applied regression analysis". John Wiley & Sons (1998).
16. F Moscato., et al. "Solution Workflows for Model-Based Analysis of Complex Systems". IEEE Transactions on Automation Science and Engineering 9.1 (2012): 83-95.
17. F Amato and F Moscato. "Model transformations of MapReduce Design Patterns for automatic development and verification". Journal of Parallel and Distributed Computing 110 (2017): 52-59.
18. F Amato, F Moscato and F Xhafa. "Generation of game contents by social media analysis and MAS planning". Computers in Human Behavior 100 (2019): 286-294.
19. Bai S, Kolter JZ and Koltun V. "An empirical evaluation of generic convolutional and recurrent networks for sequence modelling". arXiv preprint arXiv:1803.01271 (2018).
20. Narwariya J., et al. "Graph neural networks for leveraging industrial equipment structure: An application to remaining useful life estimation". arXiv preprint arXiv:2006.16556 (2020).
21. Zhou H., et al. "Informer: Beyond efficient transformer for long sequence time-series forecasting". In Proceedings of the AAAI conference on artificial intelligence 35.12 (2021): 11106-11115.
22. Wu H., et al. "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting". Advances in neural information processing systems 34 (2021): 22419-22430.
23. Oreshkin BN., et al. "N-BEATS: Neural basis expansion analysis for interpretable time series forecasting". arXiv preprint arXiv:1905.10437 (2019).
24. Challu C., et al. "Nhits: Neural hierarchical interpolation for time series forecasting". In Proceedings of the AAAI conference on artificial intelligence 37.6 (2023): 6989-6997.
25. Salinas D., et al. "DeepAR: Probabilistic forecasting with autoregressive recurrent networks". International journal of forecasting 36.3 (2020): 1181-1191.

26. Lim B., et al. "Temporal fusion transformers for interpretable multi-horizon time series forecasting". International Journal of Forecasting 37.4 (2021): 1748-1764.
27. Arena F., et al. "Predictive maintenance in the automotive sector: A literature review". Mathematical and Computational Applications 27.1 (2021): 2.
28. Gasthaus J., et al. "Probabilistic forecasting with spline quantile function RNNs. In The 22nd international conference on artificial intelligence and statistics". PMLR (2019): 1901-1910.
29. Sasinthiran A, Gnanasekaran S and Ragala R. "A review of artificial intelligence applications in wind turbine health monitoring". International Journal of Sustainable Energy 43.1 (2024): 2326296.
30. Steckel J., et al. "Tool Wear Prediction in CNC Turning Operations using Ultrasonic Microphone Arrays and CNNs". arXiv preprint arXiv:2406.08957 (2024).
31. Tang Y, Zhang Y and Li J. "A time series driven model for early sepsis prediction based on transformer module". BMC Medical Research Methodology 24.1 (2024): 23.
32. Tian Y., et al. "Federated vehicular transformers and their federations: Privacy-preserving computing and cooperation for autonomous driving". IEEE Transactions on Intelligent Vehicles 7.3 (2022): 456-465.
33. Gupta S and Torra V. "Differentially Private Traffic Flow Prediction Using Transformers: A Federated Approach. In European Symposium on Research in Computer Security". Cham: Springer Nature Switzerland (2023): 260-271.
34. Prechelt L. "Early stopping-but when?". In Neural Networks: Tricks of the trade. Berlin, Heidelberg: Springer Berlin Heidelberg (2002): 55-69.
35. Arlot S and Celisse A. A survey of cross-validation procedures for model selection (2010).
36. Al-Kababji A, Bensaali F and Dakua SP. "Scheduling techniques for liver segmentation: Reducelronplateau vs onecyclelr". In International conference on intelligent systems and pattern recognition. Cham: Springer International Publishing (2022): 204-212.
37. Li L., et al. "Hyperband: A novel bandit-based approach to hyperparameter optimization". Journal of Machine Learning Research 18.185 (2018): 1-52.
38. Snoek J, Larochelle H and Adams RP. "Practical bayesian optimization of machine learning algorithms". Advances in neural information processing systems 25 (2012).
39. Jaderberg M., et al. "Population based training of neural networks". arXiv preprint arXiv:1711.09846 (2017).
40. Izmailov P., et al. "Averaging weights leads to wider optima and better generalization". arXiv preprint arXiv:1803.05407 (2018).
41. Smith LN. "Cyclical learning rates for training neural networks". In 2017 IEEE winter conference on applications of computer vision (WACV). IEEE (2017): 464-472.
42. Mukherjee K, Khare A and Verma A. "A simple dynamic learning rate tuning algorithm for automated training of DNNs". arXiv preprint arXiv:1910.11605 (2019).
43. Goodfellow I., et al. "Deep learning". Cambridge: MIT press 1.2 (2016).