SHAP-Based Feature Selection with a Hybrid Convolutional and Recurrent Deep Learning Framework for Remaining Useful Life Prediction of Aero-Engines

Naouel Ouafek [1], Tarek Berghout [2], Samira Abderrezek[3], and Abdelhabib Bourouis [4]

[1,3] *Ecole Normale Superieur de Constantine,Constatine, Algeria*
ouafek.naouel@ensc.dz
Abderrezak.samira@ensc.dz

[2] *Laboratory of Automation and Manufacturing Engineering University of Batna 2,05000, Batna, Algeria*
t.berghout@univ-batna2.dz
[4] *ReLa(CS)2 Laboratory, University of Oum El Bouaghi, P.O. Box 358 04000, Oum El Bouaghi ,Algeria*
bourouis.abdelhabib@univ-oeb.dz

ABSTRACT

Prognostics and health management for aeroengines is crucial, especially for reducing catastrophic loss of life and minimizing maintenance costs. Accurate Remaining Useful Life (RUL) estimation optimizes schedules, lowers costs, and enhances safety. Physics-based modeling for RUL assessment is challenging due to its inability to fully account for system- and environment-related dynamics. Therefore, machine and deep learning approaches are highly recommended. This paper proposes a new method for RUL prediction of turbofan engines using the well-known (CMAPSS) dataset. Generally, data from acquired engines may be corrupted by anomalies due to sensor failures or environmental disturbances, which can affect the accuracy of prediction models. Therefore, this paper combines advanced techniques for reducing irrelevant and redundant sensor signals by applying Shapley Additive Explanations (SHAP) to refine feature selection by quantifying each sensor's contribution to RUL prediction, ensuring both interpretability and efficiency. Then an anomaly detection and removal followed by RUL prediction with deep learning are used for such complex tasks. Specifically, K-means clustering, autoencoders, Temporal Convolutional Networks (TCN) and Long Short-Term Memory (LSTM) networks. This approach enables effective data preprocessing by detecting and removing anomalies, thus enhancing the quality of the training data. Moreover, uncertainty analysis is performed to assess the reliability of the predictions, including confidence intervals for the results. The experimental results show substantial improvements in predictive accuracy, confirmed by better evaluation metrics, numerical evaluations, and the coefficient of determination, compared to traditional approaches. The TCN-LSTM model achieved an average performance improvement of 24.05% on FD001, 12.44% on FD003, and 4.63% on FD002 (using Monte-Carlo uncertainty estimation). Furthermore, for the FD004, a performance decrease of -4.68

https://doi.org/10.36001/IJPHM.2026.v17i1.4693

1. INTRODUCTION

Prognostics and health management are crucial for enhancing system reliability, ensuring safety, and minimizing operational costs. In complex systems like aircraft engines, predictive maintenance significantly contributes to longevity and efficient operation. Traditional methods, such as physics-based models, have been used for RUL prediction (Arias Chao, Kulkarni, Goebel, & Fink, 2022) . However, these models often struggle to fully capture the complexities of real-world systems, as they cannot accurately account for all system and environment-related dynamics (Li, Zhang, Li, & Si, 2024). As a result, these methods face limitations in handling the intricate, non-linear nature of degradation processes. In response, there has been a shift towards hybrid models that combine physics-based approaches with data-driven techniques (Berghout & Benbouzid, 2022; Lei, Li, Guo, Yan, & Lin, 2018). These models leverage the strengths of both paradigms and are complemented by deep learning algorithms, also known as representation learning, which excel in capturing complex patterns in large, high-dimensional data. Accurate RUL prediction is of great importance to avoid future failure. Such multivariate time series data exhibit complex spatiotemporal dynamics, and their analysis requires sophisticated modeling. Traditional methods usually fail to simultaneously cap-

ture such dependencies, resulting in suboptimal predictions.
More specifically, modern representation learning techniques for RUL prediction utilize advanced hybrid methods that address the selection of informative sensors, complexity, and uncertainty inherent in sensor data. These techniques abstract the data at different stages, capturing both linear and nonlinear relationships, which enhances the model's ability to handle dynamic system behaviors. By integrating physics-based models with data-driven approaches, these models can better represent system degradation and improve the accuracy and robustness of RUL predictions, especially under variable operating conditions.
Classical approaches include correlation analysis, PCA, and autoencoder-based dimensionality reduction (Lei et al., 2018). These methods reduce redundancy and noise, but they are essentially statistical, making it difficult to justify why specific sensors are kept or removed in practice.
On the other hand, sensors often contain outliers and abnormal readings that negatively affect performance. Statistical methods such as Z-score and moving averages (Hawkins, 1980) have been widely used to detect and replace anomalies, but they struggle with complex nonlinear anomalies. More sophisticated approaches based on clustering, such as K-means (Chandola, Banerjee, & Kumar, 2009), or on reconstruction-based detection with autoencoders (Sakurada & Yairi, 2014). Hybrid methods, which combine clustering and autoencoder reconstruction error, have shown strong performance by enabling both global and local anomaly detection (H. Xu et al., 2019). Such anomaly handling improves robustness and generalization in RUL estimation tasks.
Concerning the RUL prediction, we found a set of interesting recent works to analyze for research contributions and gaps. For instance, in (Deng & Zhou, 2024c), the authors proposed a hybrid approach, such as Convolutional Neural Network - Long Short-Term Memory (CNN-LSTM) - Attention, where spatial characteristics are extracted by CNNs, temporal dependencies are captured by LSTMs, and attention mechanisms further enhance the accuracy of prediction. In (Liu, Wang, & Yu, 2021c), the authors present a combination of Deep CNN for feature extraction and LightGBM for fast predictions. The authors showed that the use of this algorithm improves the prediction accuracy of traditional models. The results of the study show promising and effective performance in predicting the RUL of aircraft engines.
Attention-based approaches, such as multi-head self-attention, mine dependencies among time series using multiple attention heads are also introduced (X. Wang et al., 2023),(X. Wang et al., 2024). In 2023, they proposed an improved multi-head self-attention model that reveals complex temporal dependencies and highlights the most discriminative features for RUL prediction for aircraft engines. In 2024, they introduced an improved feature selection approach including temporal attention mechanisms. This method focuses on temporal relationships to perform improved feature selection, thus increasing the prediction performance. In (Deng & Zhou, 2024b),(J. Hu, Jiang, Xu, & Zhang, 2025), investigated the use of Recurrent Neural Network (RNN) models, such as LSTM and Bi-LSTM based models. In their first study, they explore the use of (LSTM) networks to capture temporal dependencies and improve which are applied to extract temporal dependencies from sensor data. In a complementary study, the authors introduce Bi-LSTM networks, with the attention mechanisms that play a crucial role in identifying the most relevant temporal features, thereby enhancing the robustness of aircraft engines remaining useful life predictions.
Moreover, references (Liu, Wang, & Yu, 2021b), (Liu, Wang, & Yu, 2021d)study boosting models, such as (LightGBM) and (XGBoost) have wide applications in large and complex datasets and are known for their speed, efficiency, and balance between speed and prediction accuracy under various conditions.
Lastly, approaches based on convolutional neural networks are considered in the process of taking advantage of the spatial characteristics of the data by using convolutional layers and focus on feature extraction using CNNs, which allow extraction of much deeper features for better performance (Deng & Zhou, 2024a), (Liu, Wang, & Yu, 2021a). The methods exhibit the diversity and complementarity of the methods available for RUL prediction, borrowing strengths from both neural networks and optimization techniques.
Despite the promising results obtained by recent deep learning-based RUL prediction models, including CNN–LSTM (Deng & Zhou, 2024c), Bi-LSTM (J. Hu et al., 2025), and attention-based architectures (Xin et al., 2023), several limitations hinder practical PHM deployment. In particular, many existing approaches focus on predictive accuracy, while challenges related to noisy sensor measurements, early degradation patterns, and the lack of predictive uncertainty quantification are often insufficiently studied. Additionally, preprocessing approaches frequently do not clearly distinguish between noise, outliers which may result in the removal of informative signals or in reduced model robustness.

Recent advances in Explainable Artificial Intelligence (XAI) have sought to address the black-box nature of deep learning models. One of these methods, Shapley Additive Explanations (SHAP) has seen increasing use due to its ability to provide consistent and quantitative estimates of feature relevance. For example, Zhang et al. (Hong et al., 2020) employed SHAP to identify relevant sensors in CNN–LSTM-based RUL models, while Li et al. (Genane & Aalah, 2023) joined SHAP with feature clustering to extract interpretable insights at the group-level. More recently, Alomari et al. (Alomari, Baptista, & Ando, 2024) integrated SHAP with deep-network based analysis to study sensor impact in aero-engine prognostics with $C-MAPSS$ dataset. These studies highlight the prospective of SHAP to enhance transparency and to build trustworthy in RUL prediction.

Table 1 presents a summary of the literature reviewed in the study. and highlights the identified research gaps. The major contributions of this work are as follows:

- **Data preprocessing:** A hybrid approach using K-means and autoencoders for handling noisy data while preserving informative degradation features.
- **Interpretable feature selection:** The combination of correlation analysis and SHAP to evaluate the relevance of the sensor data in an interpretable manner.
- **Uncertainty-aware RUL prediction:** An $TCN-LSTM$ with Monte Carlo dropout.
- **Improved prognostic performance:** The experimental results on the C-MAPSS dataset indicate a reduction in predictive errors and competitive results in terms of MAE, RMSE, and R^2.

The remainder of this paper is organized as follows: Section 2 presents detailed description of the datasets used in this study, as well as the data preparation, Section 3 discusses the proposed methodology, Section 4 presents a discussion of the results, and compares our model's performance against other approaches in the literature. Section 5 concludes the paper and highlights potential avenues for future research.

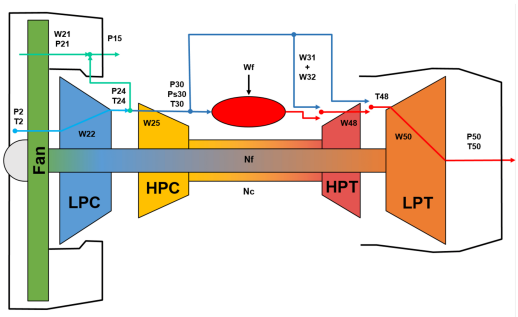


Figure 1. Schematic diagram of the CMAPSS turbofan engine.

This figure is reproduced from (Arias Chao, Kulkarni, Goebel, & Fink, 2021), under open access license.

1.1. Dataset description

As mentioned earlier, this study uses the CMAPSS data set, a well-known benchmark for turbofan engine degradation analysis. Originally released by NASA in 2008 as part of the Prognostics and Health Management Society Challenge (Saxena, Goebel, Simon, & Eklund, 2008), the dataset is widely recognized for its reliability and utility in predictive maintenance research. It is named after the CMAPSS (Commercial Modular Aero-Propulsion System Simulation) software, developed on the MATLAB platform, which is specifically designed to study and analyze engine performance under varying flight conditions. Figure (1) depicts a schematic of the turbofan model used in the simulations studied in this work. The CMAPSS dataset provides complete engine degradation trajectories, capturing the progression from healthy states to failure. It is divided into four subsets, FD001 to FD004, see Table (2) (Saxena et al., 2008), which represent different operating and failure modes. Each subset includes a training set, a test set, and actual RUL values, enabling thorough validation of predictive models. The division of subsets is based on the observed (injected) failure types during the simulation process. The dataset consists of readings from 21 sensors that monitor critical engine parameters and 14 modifiable input parameters used to simulate varying operating conditions. This highly realistic model accounts for variability induced by manufacturing differences and initial wear. The primary objective is to predict the RUL of engines for each subset and compare the predictions against the RUL labels in the test data. Notably, the number of life cycles Figure 2 shows the statistical distribution (mean and standard deviation) of all sensor readings in the dataset.

1.2. Data processing

The CMAPSS data are provided as a zip-compressed text file with 26 columns of numbers separated by spaces. Each row is a snapshot of data taken during a single operational cycle, and each column represents a different variable. The data is contaminated with sensor noise (Saxena et al., 2008).

1.2.1. Feature selection

Some of the parameters measured by turbofan sensors are poorly correlated with RUL. Those sensors are not very informative concerning degradation analysis and must be discarded to increase the efficiency of the network training process.
A correlation analysis and feature relevance is used to eliminate irrelevant or redundant features.
In addition to correlation analysis, an explainability-driven refinement step is used. SHAP values were calculated on a preliminary deep learning model (BiLstm) that had been trained on all available sensors (only training data). SHAP values are the contribution value of each sensor to the task of RUL prediction. Features that universally had low SHAP importance and low correlation with RUL were removed. This integration of XAI-based analysis and statistical correlation guarantees that the final input variable set is not only statistically significant but also interpretable in relation to the prediction model. Figure (3) illustrates a sample SHAP feature importance for the FD001 subset.

Table 4 summarizes the results of the explainable Shap and the correlation matrix as indicated in Figure (4, 5). Sensors such as Total temperature at fan inlet (T2), Pressure at fan in-

Table 1. Summary of previous works in terms of datasets used, methods, advantages, and limitations.

Ref	**Methods**	**Dataset**	**Advantages**	**Limitations**
(Deng & Zhou, 2024c)	CNN_LSTM _Attention, data augmentation, noise reduction	Data from industrial machinery	Captures spatial and temporal dependencies	High computational cost, requires extensive data preprocessing
(Liu et al., 2021c)	CNNs, LightGBM, feature engineering, normalization	Data from manufacturing processes	Fast predictions, handles missing values	Limited ability to capture complex temporal dependencies
(X. Wang et al., 2023),(X. Wang et al., 2024)	Multi-head self-attention, data cleaning, imputation	Time series data from IoT devices	Captures dependencies across time steps, improves accuracy	Requires careful tuning of attention mechanisms, sensitive to hyperparameters
(Deng & Zhou, 2024b), (J. Hu et al., 2025)	Bi-LSTM, data preprocessing, dimensionality reduction	Data from transportation systems	Captures forward and backward dependencies, reduces uncertainty	Sensitive to data preprocessing techniques, high computational requirements
(Arias Chao et al., 2022)	Fusion of physics-based and deep learning models	CMAPSS dataset	Enhances prediction accuracy and robustness	Requires domain knowledge for model fusion, high computational
(Berghout & Benbouzid, 2022)	Systematic guide for machine learning models	Machinery datasets	Comprehensive approach to RUL prediction	May be complex for beginners, requires extensive data preprocessing
(Deng & Zhou, 2024a)	CNN for feature extraction	CMAPSS dataset	Effective feature extraction	Require large computational resources
(Asif et al., 2022)	Deep learning model	CMAPSS dataset	Accurate RUL prediction	Requires large datasets, high computational cost
(S. Wang, Ji, Wang, Ma, & Chen, 2022)	Deep convolutional LSTM network	CMAPSS dataset	Captures temporal dependencies	Computationally intensive, requires large datasets
(Fan, Li, & Chang, 2023)	Bi-LSTM autoencoder transformer	Various datasets	Accurate RUL estimation	Requires extensive training, high computational cost
(Tan & Teo, 2021)	Temporal convolution with attention	Various datasets	Enhances feature selection	Require careful tuning,sensitive to hyperparameters
(Xin et al., 2023)	Time-enhanced multi-head self-attention model	CMAPSS dataset	Enhances feature selection	Requires careful tuning, sensitive to hyperparameters
(Ayodeji, Wang, Su, Yuan, & Liu, 2021)	Attention-based multi-head models	CMAPSS dataset	Improves RUL prediction	Requires extensive hyperparameter tuning, high computational cost

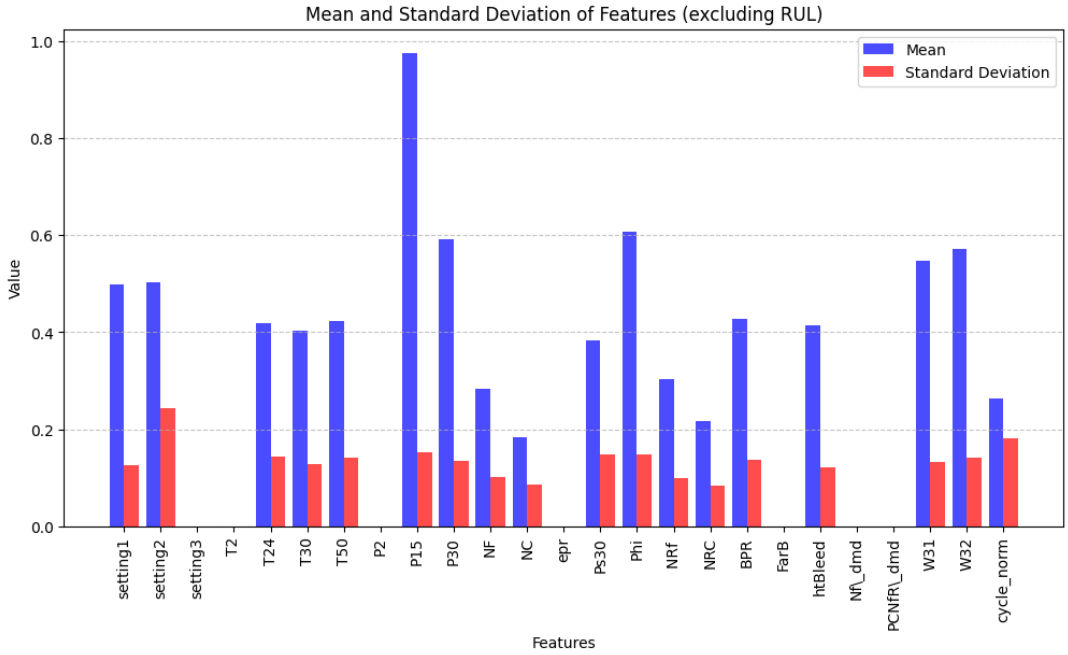


Figure 2. Descriptive Statistics of Variables.

Table 2. Composition of the CMAPSS dataset

Dataset	Fault Modes	Train	Test
FD001	ONE (HPC Degradation)	100	100
FD002	ONE (HPC Degradation)	260	259
FD003	TWO (HPC ,Fan Degradation)	100	100
FD004	TWO (HPC ,Fan Degradation)	248	249

let (P2), Total pressure in bypass-duct (P15), Engine pressure ratio (epr), Burner fuel-air ratio (FarB), Demanded fan speed (Nf_dmd) and Demanded corrected fan speed (PCNfR_dmd) are not strongly related to the degradation process; hence, these sensors were also excluded from the input features of the proposed method for FD001.

In the FD003 dataset, from the correlation analysis and by following (Asif et al., 2022), 16 sensors were retained that included those used for FD001 plus two sensors P15 and epr because they are correlated with RUL values. For the remaining two data subsets, FD002 and FD004, all sensor measurements were retained since they indicated a strong correlation with RUL.

However, the differences in the level of importance of the sensors between the various subsets of the CMAPSS are primarily due to the differences in the conditions of operation, the types of faults, and the degradation trends rather than the differences in the engine configuration. The impact of this selection on FD001 and FD003 with the removal of some unnecessary sensors is to reduce noise and improve convergence. For FD002 and FD004, all sensors were kept because they had a high correlation with RUL, indicating that in complex scenarios, a larger amount of information can be beneficial. Once the irrelevant features have been removed, the sensor readings are then normalized for consistent input scales for deep learning models.

-Based Feature Selection: (Ablation Study)

To assess the contribution of the proposed SHAP-based feature selection strategy, an ablation study is carried out by comparing two input configurations. In the first configuration, all available sensors are included without any feature selection process. Another configuration includes the proposed SHAP-based feature selection process for the selection of only relevant sensors based on their effectiveness in RUL prediction. For both configurations, the same baseline deep learning architecture (BiLSTM) and identical training protocol are employed to ensure a fair and controlled comparison. The performance of the models is measured using three metrics: Root Mean Squared Error (RMSE), R^2 score, and penalty score. The numerical results before and after SHAP-based feature selection are shown in Table 3. The results indicate that incorporating SHAP-based feature selection leads to consistent performance improvements. In fact, the performance improvement can be clearly identified by the decrease

Table 3. Ablation study of SHAP-based feature selection on the C-MAPSS dataset

FD001	RMSE	R^2 Score	Penalty Score
Before SHAP Selection	16.59	0.82	342.24
After SHAP Selection	14.26	0.87	217.21

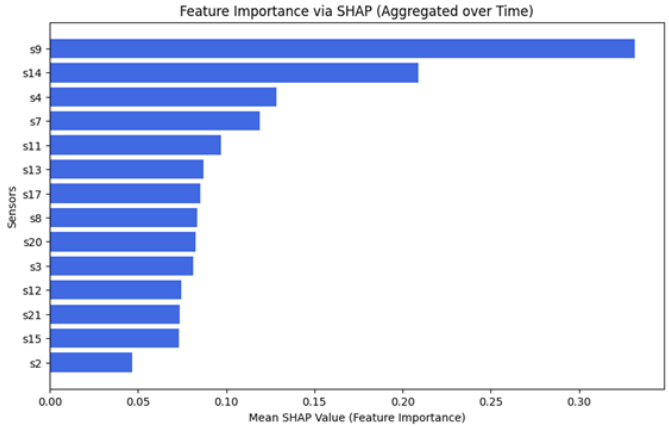


Figure 3. SHAP results.

in the value of the Root Mean Squared Error (RMSE) from 16.59 to 14.26 and the increase in the R^2 score by 0.87 compared to 0.82 for the prediction model. In the same way, the penalty score has decreased by 342.24 to 217.21, indicating enhanced robustness in RUL estimation. These results demonstrate that selecting relevant sensors based on SHAP not only improves predictive performance but also contributes to a more stable and interpretable modeling process.

1.2.2. Data normalization

The various characteristics of data usually come in different scales and units, which might influence the results while analyzing the raw data. It is important to normalize data to reduce the effects of such differences, thus providing a chance for comparison between features on common ground. For this work, the normalizing method used is min max normalization as in Equation (2), where x' and x are the normalized and original measurements, respectively.

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{1}$$

2. PROPOSED METHODOLOGY

Combining advanced anomaly detection and rectification with the deep learning-based framework for understanding complex dynamics that underline time series data, this work will further propose an overall methodology divided into three main phases: data preparation, modeling, and uncertainty estimation as shown in figure 6.

2.1. Cluster-Based Outlier Handling

Data quality has a major influence on predictive model results, especially in complex predictive models like predictive maintenance. In this work, a clear distinction is made between *outliers*, *noise*, and *degradation-related patterns*. Outliers refer to sporadic and non-physical measurements col-

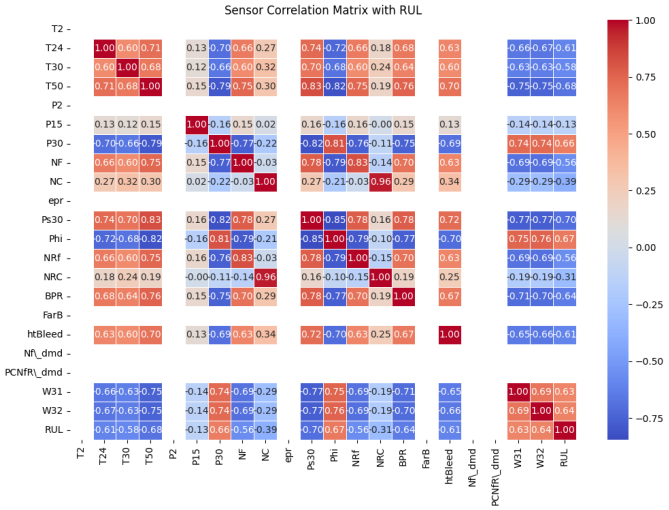


Figure 4. Correlation Matrix of FD001's sensor.

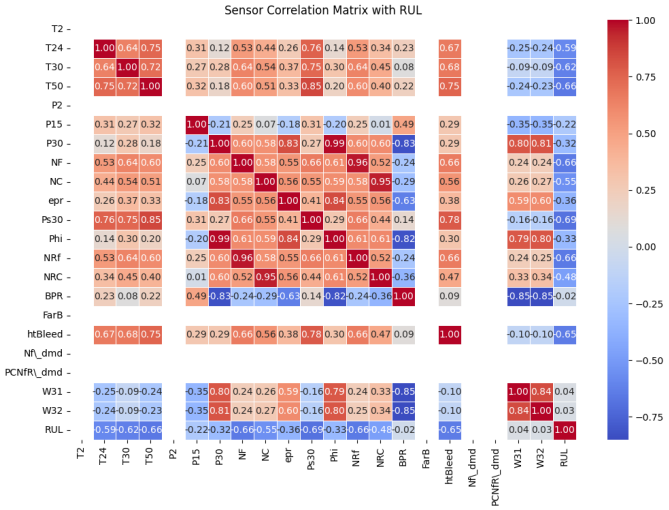


Figure 5. Correlation Matrix of FD004's sensor.

Table 4. Sensor Evaluation Based on Correlation and SHAP Importance

Sensor	Correlation with RUL	SHAP Importance	Justification	Decision
T2(s1)	∼0.63	Low	Moderate correlation but low SHAP importance.	Not Retained
T24(s2)	∼-0.61	Very Low	Moderate correlation and low SHAP.	Retained
T30(s3)	∼0.60	Low	Moderate correlation, low importance in SHAP.	Retained
T50(s4)	∼0.70	High	Strong correlation + strong SHAP.	Retained
P2(s5)	∼0.30	Very Low	Weak correlation and no SHAP importance.	Not Retained
P15(s6)	∼0.13	–	No contribution from SHAP; very weak correlation.	Not Retained
P30(s7)	∼-0.66	Medium-High	Strong negative correlation, moderate SHAP.	Retained
Nf(s8)	∼0.63	Medium	Good correlation + decent SHAP.	Retained
Nc(s9)	∼0.34	Very High	a high SHAP value despite weak correlation	Retained
epr(s10)	∼0.34	–	Weak correlation, not important in SHAP.	Not Retained
Ps30(s11)	∼0.72	Medium	Strong correlation and fair SHAP.	Retained
phi(s12)	∼-0.70	Low	Strong negative correlation, but low SHAP.	Retained
NRf(s13)	∼0.70	Medium	Strong correlation + good SHAP.	Retained
NRc(s14)	∼0.67	High	Strong correlation + high SHAP.	Retained
BPR(s15)	∼0.70	Low	Strong correlation, but low SHAP.	Retained
farB(s16)	∼0.29	–	Weak correlation, no SHAP relevance.	Not Retained
s17(htBleed)	∼0.39	Medium	Moderate correlation, moderate SHAP.	Retained
Nf_dmd(s18)	∼-0.64	–	Strong negative correlation, but it is a Constant value.	Not Retained
PCNfR_dmd(s19)	∼0.64	–	Decent correlation, but it is a Constant value.	Not Retained
PCNcR_dmd(s20)	∼0.67	Medium	Good correlation and decent SHAP.	Retained
W31(s21)	∼0.64	Low	Acceptable correlation, weak SHAP.	Retained

lected through sensors due to acquisition errors or external disturbances. Conversely, degradation patterns relate to physical and constant variations that affect sensors due to system aging. The proposed preprocessing strategy is designed to mitigate noise and isolate spurious outliers, while explicitly preserving degradation-related information that is essential for RUL modeling. The approach integrates clustering techniques with autoencoder-based reconstruction and consists of two main stages: First, data clustering and then noise and outlier detection in each cluster. Autoencoders are widely used for anomaly detection; however, In this work, the use of autoencoders was redefined for the purpose of mitigating noisy data rather than the detection of failures through the use of the anomaly filter. By incorporating the use of the K-means clustering technique for preprocessing the data, the model was able to recognize the homogeneous pattern of the data before the use of the autoencoders for the purpose of reconstructing the CMAPSS data set.

2.1.1. Automatic Detection of Clusters in Data using K-means

To perform this clustering we found a variety of methods such as K-means, DBSCAN and Isolation Forest. DBSCAN does not scale well to large datasets and may poorly handle clusters of complex shapes. Isolation Forest can only detect global anomalies; structural relationships of the time series are not considered. K-means is well-established method for data segmentation (MacQueen, 1967), (R. Xu & Wunsch, 2005) and has been successfully applied in RUL prediction to enhance preprocessing steps (Arias Chao et al., 2022).

K-means were chosen for simplicity and efficiency of computation and to allow for more accurate anomaly detection, as this technique adapts to the characteristics of subsets of the CMAPSS,and the purpose of clustering is done in a way that groups sensors that have similar statistical behavior, rather than requiring that clusters be separable. Therefore, K-means clustering is used to divide the data into homogeneous groups, allowing a reduction in the investigation of anomalies in each obtained group.
K-means inputs are raw sensor data without any dimensionality reduction via PCA. This choice enables us to keep the global information of the dataset, preserving along the clustering the natural data structure. Since each CMAPSS subset represents various operating conditions and failure modes, K-means clustering learns dynamically by forming homogeneous clusters according to the specificities of each dataset. This allows for a more precise distinction between sensor behaviors before detecting anomalies.

a) Determining the Optimal Number of Clusters In order to determine the optimal number of clusters k, we tested values from 2 to 15 and took into account two significant metrics:

Elbow Method : which analyzes the reduction of inertia as a function of the number of clusters. The optimal $k_{optimal}$ is where adding more clusters does not significantly reduce inertia.

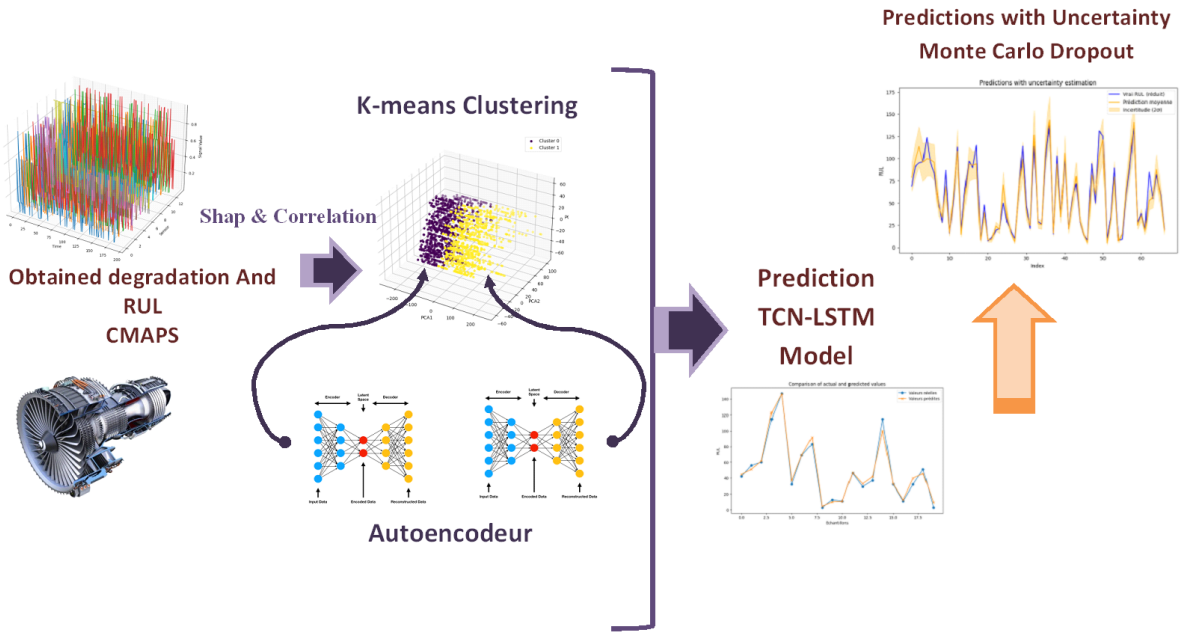


Figure 6. Flowchart of the proposed methodology.

Silhouette Score – Measures inter-cluster separation and intra-cluster cohesion. Silhouette Score above 0.5 indicates good clustering.

The Elbow and Silhouette Score analysis outcomes are graphed in Figure (7b). The optimal number of clusters varies according to the dataset subset. For instance, for FD001, the optimal k according to the Silhouette Score was 2. The dataset is then clustered accordingly, and each sample is labeled with a cluster number, which is added as a new feature for future anomaly detection (Figure (7a)).

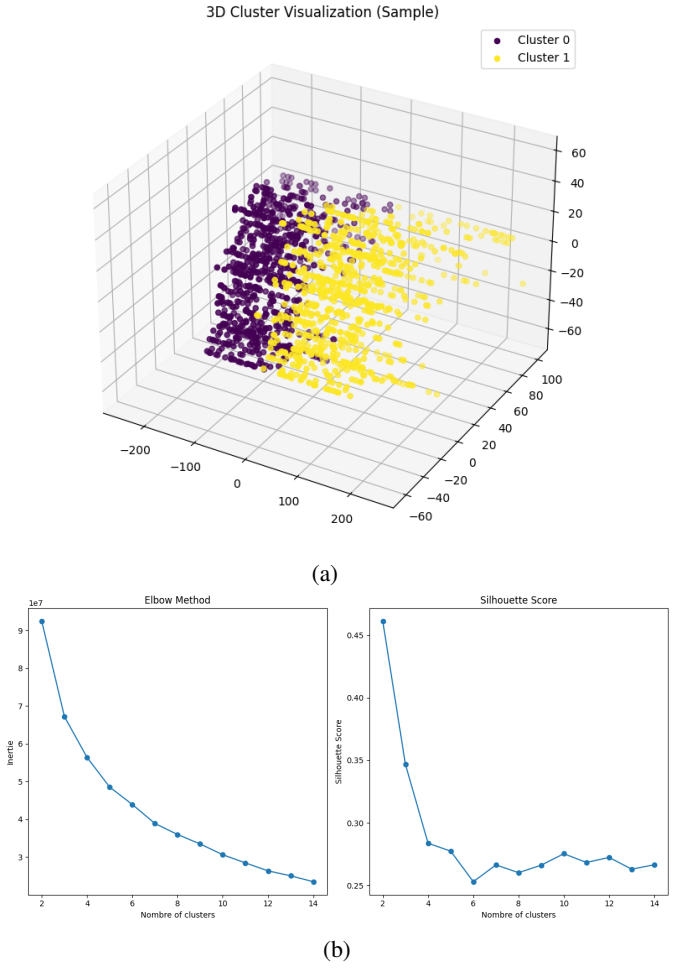
b) Clustering Performance Evaluation Silhouette Score was computed for different values of k on subsets of CMAPSS. The results confirmed that the optimal number of clusters ranged from 2 to 5, with a clear grouping of data points.

The selected clusters ensure that the behavior of every sensor is well-separated, which reduces noise before anomaly detection. The approach enhances data quality while adapting dynamically to different operating modes in CMAPSS. RUL of aircraft engines

2.1.2. Autoencoder for outliers detection

An autoencoder is trained separately for each previously obtained cluster to detect anomalies based on reconstruction errors. The architecture of the used autoencoder is presented in figure (8). It contains an input layer and the dimensionality of the inputs depends on the number of sensors in the dataset. In our case, each input contains N features, corresponding to the measurements of the engine sensor. In the Encoding: Three dense layers, with decreasing dimension, ReLU activations, and Dropout (0.2) and regularization to avoid over-fitting. In the Decoding: The inverse symmetric layers reconstruct the original data.

To perform anomaly detection using this autoencoder, reconstruction errors are calculated for each data in the cluster after training. The data for which the error exceeds the 95th percentile of the reconstruction errors are considered anomalous. This adaptive way of setting the threshold allows the particular distribution of errors for each cluster to be taken into account. The 95th percentile is commonly used in outlier detection as it effectively captures extreme values while reducing the impact of normal variations.
Once the anomalies are detected, a replacement and correction phase is carried out based on two complementary techniques used to correct these anomalies, the first one is Rolling Mean which consist of moving window to smooth the data, providing a local correction sensitive to trends.A 5-point moving window was employed since it provides a good balance between reducing noise and maintaining major variations. This selection is in line with prior research on correcting noisy time series. The second one is Z-score which considers a data point with an absolute Z-score greater than 3 is used to be extreme anomalies. This value is generally used in anomaly detection to remove outliers that are too extreme without per-



(a)

(b)

Figure 7. a) 3D k-means results for a sample of FD001 (2 clusters), b) Automatic selection of clusters.

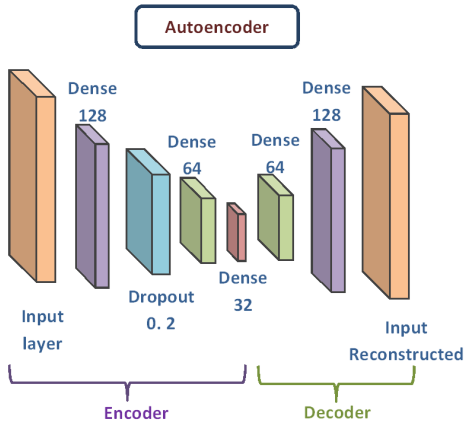
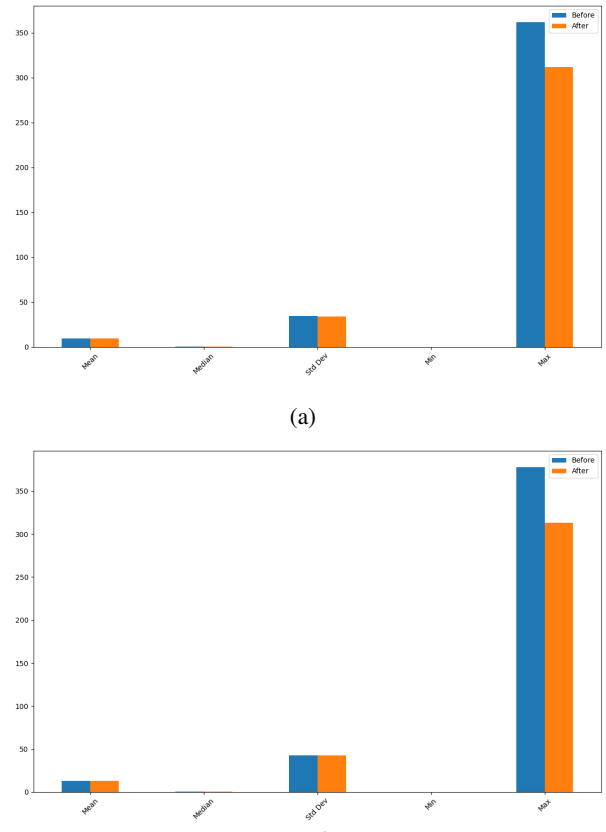


Figure 8. The used autoencoder.



(a)

(b)

(c)

Figure 9. Comparison of statistics before and after correction of anomalies in a) FD001, B)FD004

Table 5. Results obtained on FD001 to FD004.

Dataset	Number of detected outliers in each subset
FD001	515
FD002	1343
FD003	620
FD004	1528

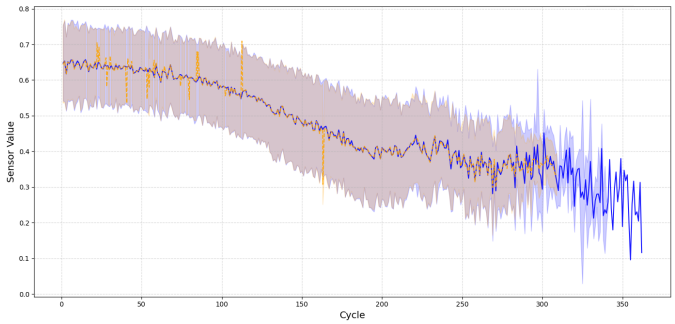
turbing the main distribution of the data. The values are then replaced by the mean of the corresponding column. This tandem application guarantees a robust correction without affecting the overall trends of the data. The application of the proposed process on the original data allows the effective detection and correction of anomalies as it is showen in the comparative Table (5). For example, in the case of the FD001 subset, 515 anomalies were detected and corrected.

The statistical results before and after correction are presented in Table (6) and figure (9) show a remarkable improvement in the consistency of the data: The reduction of the standard deviation from 34.71 to 34.21 reflects a significant reduction in the dispersion of the data indicating an effective attenuation of outliers. The reduction of the maximum value from 362 to 312 concerns the correction of extreme values that had a strong influence on the overall statistics. The mean and median did not change much after correction. Therefore, the adjustment performed maintained the central tendency of the data without significant distortion.

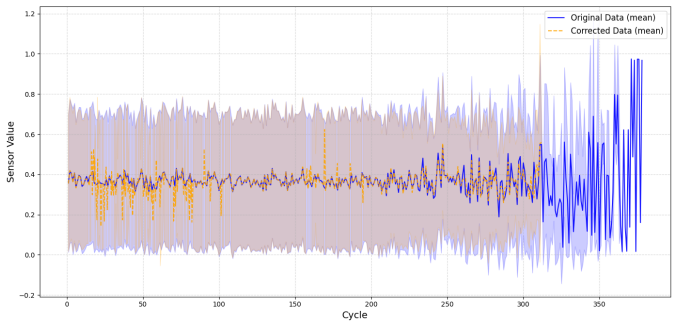
These results demonstrate the ability of the proposed method to improve the quality of the data while preserving their underlying structure as shown in Figure(10). The increase in peaks in the corrected data curve (orange line) compared to the original data is caused by the nature of the Rolling Mean and the Z-score. The Rolling Mean may accentuate some local oscillations, especially when there are outliers in small time periods, and the Z-score, by replacing the outliers with the mean, can overemphasize some inherent oscillations. Upon close examination, one sees that these new peaks occur in locations where anomalies used to be present frequently, which means that the correction has been successful in identifying and substituting the outliers. Secondly, the confidence interval is the range of the values reconstructed by the autoencoder after correction: a high range means high uncertainty, while a low range means a more stable and reliable correction.

Table 6. Statistics before and after correction of anomalies for FD001.

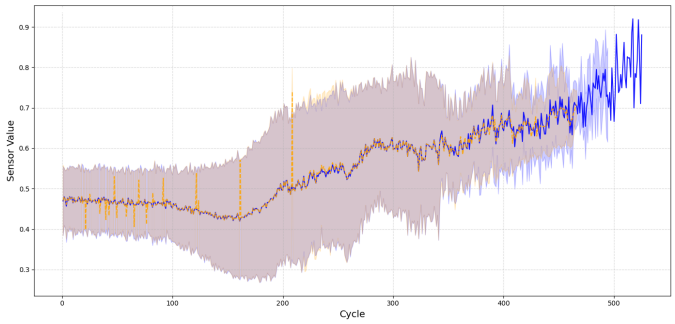
Statistics	**Before Correction**	**After Correction**
Mean	9.866871	9.786299
Median	0.366197	0.367470
Std Dev	34.711151	34.213132
Min	0.000000	0.000000
Max	362.000000	312.000000



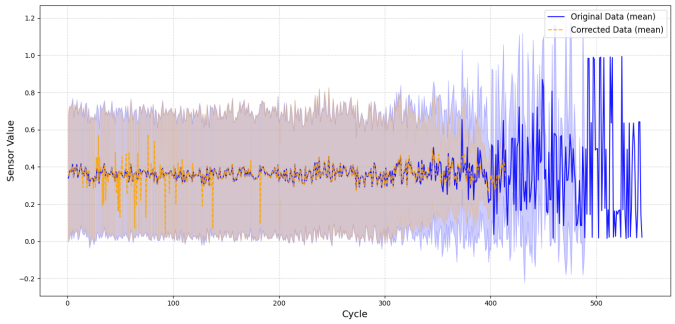
(a)



(b)



(c)



(d)

Figure 10. Illustration of original (blue) and corrected data (orange): the Mean and Confidence Intervals for Sensor:W32 in a) FD001, B)FD002, c)FD003, D)FD004.

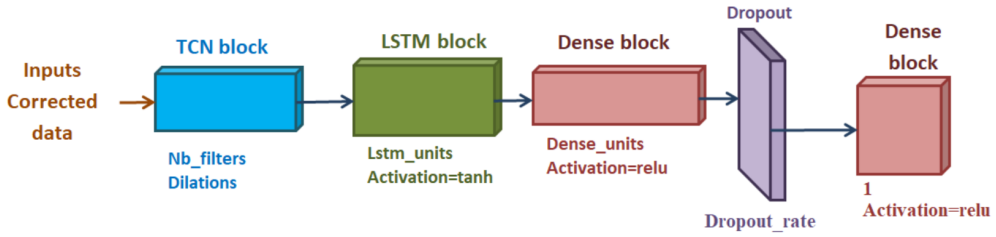


Figure 11. TCN-LSTM Model.

2.2. TCN-LSTM Model for Rul prediction

After outliers were detected and corrected, we were faced with a variety of deep learning models (X. Wang, Zhao, & Pourpanah, 2020), including those specific to time series problems (Kong, Chen, & Liu, 2025). After careful study, two models were combined to build the architecture used in the proposed methodology. We propose an architecture based on temporal convolutional networks **TCN** combined with Long-Short-Term Memory networks **LSTM**. This combination model can capture both long-range sequential dependencies due to TCN and complex dynamics of multivariate time series processed by the LSTM. The TCN-LSTM model has demonstrated its applicability to a wide range of fields, like in (T. Hu, Zhu, & Tian, 2025) study the optimization of the reuse and sustainability of electric vehicle batteries by predicting their RUL. In (C. Hu, Cheng, Ma, & Li, 2022) study uses this model to estimate the state of charge of lithium-ion batteries, contributing to efficient energy management in storage systems.

In air pollution prediction: Deep learning models such as TCN-LSTM are used to predict pollutant concentrations like PM10 and PM2.5 for use in environmental monitoring and prevention of effects (Sharma, Singh, & Choudhary, 2025).

These solutions demonstrate the competence of TCN-LSTM as a problem solver in various fields, including energy systems, environmental science, and predictive maintenance.

2.2.1. Model Architecture

The proposed model is primarily built with TCN and LSTM networks, where each block learns from the outputs of the previous block, as summarized in Equation 2. This architecture is specifically optimized for multivariate time series prediction. The various components of the architecture model are illustrated in figure (11). :

$$\text{RUL} = f_{\text{Output}}\Big(f_{\text{Dense}}\big(f_{\text{LSTM}}\big(f_{\text{TCN}}(\mathbf{X})\big)\big)\Big) \tag{2}$$

Let denote the inputs $\mathbf{X} \in \mathbb{R}^{N \times S_l \times NB_f}$, consists of a multivariate time series data where: N is the batch size, S_l is the number of time steps in the input sequence, and NB_f is the number of features per time step. Preliminary TCN is adopted to extract the global temporal features. A convolution operation is applied to the input data with NB_f filters, a kernel size of K_s, and dilations $d = \{1, 2, 4, 8, \ldots\}$ for capturing temporal dependencies across multiple scales. For each dilation d, the output is described in Equation 3 where W_i^d: weight of the filter for dilation, $b^{(d)}$ is the bias associated with the dilated filter and the ReLU activation function.

$$\mathbf{H}^{(d)}[t] = \text{ReLU}\left(\sum_{i=0}^{K_s-1} \mathbf{W}_i^{(d)} \cdot \mathbf{X}[t - i \cdot d] + \mathbf{b}^{(d)}\right) \tag{3}$$

The outputs of the TCN are fed into an LSTM block with NB_{lstm} the number of neurons and $tanh$ the hyperbolic tangent activation functions, which is utilized to model complex temporal dependencies.
Apart from the fact the LSTM also captures dynamic interactions compared to the TCN, the multivariate features are involved, and the LSTM acts as a selective memory, deciding which information to keep, forget, or use at each time step. This block models extended sequences and complex temporal features by maintaining long-term sequential relationships capturing dynamic information that dilated convolutions might miss. The LSTM uses its internal gates defined in 4 where $\mathbf{o}_t$ is the output state of the output gate, which controls what information from the cell state is passed to the output and $\mathbf{c}_t$ is the cell states witch relevant information over the long term.

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh\left(\mathbf{c}_t\right) \tag{4}$$

To transform features extracted from TCN and LSTM blocks into accurate RUL predictions, a dense layer is used defined in Equation 5 while W_{d1},b_{d1} represent weight matrices and biases of dense layers, respectively, with the ReLU activation function. This layer is followed by a dropout layer is used with a rate of between 10% à 40% to prevent overfitting.
Finally, the output layer a dense layer with linear activation

Table 7. The Hyper-parameters values.

Variable	FD001	FD002	FD003	FD004
S_l	110	110	110	120
Input Channels	18	25	20	25
Nb_f	64	128	128	256
Dilations	1, 2, 4, 8, 16	1, 2, 4, 8	1,2	1, 2, 4
k_s	3	3	3	3
NB_{lstm}	256	256	500	128
NB_{dense}	128	100	256	150
dropout_rate	0.3	0.1	0.1	0.4

produces the RUL prediction.

$$\mathbf{H}_{\text{dense}} = \text{ReLU}(\mathbf{W}_{d1}\mathbf{H}_{\text{LSTM}} + \mathbf{b}_{d1}) \tag{5}$$

2.3. Model Configuration

There is no universal set of optimal parameters for the entire data set. The best parameter values vary for each subunit (Asif et al., 2022). Hyperparameters were optimized via a cross-validation search, combining manual tuning and grid search. Different configurations were tested for sevral parameters, and the best results were selected based on RMSE.

Table 7 displays the settings for the TCN-LSTM model instance. This includes several key parameters such as the input sequence length S_l, the number of features NB_f, kernel size K_s, dropout_rate, and number of neurons in each layer. The important parameters are presented in table 7 with the optimal values to ensure optimal performance. Tuning of hyperparameters is commonly used in RUL prediction in line with dataset variations, although it is suggested that a uniform configuration should be used in deployment-oriented applications.

2.3.1. Compilation and Training Loss Function:

The loss used is the Mean Square Error-MSE, suitable for regression tasks. Optimizer: Use Adam Optimizer with a different learning rate value for each subset for stability and efficiency. The model was trained on 150 epochs, with a batch size of 32, using cross-validation with 20% of the data. Early stopping is also implemented to avoid overfitting.
These experiments were performed on Google Colab Pro, which offers a Tesla T4 GPU with 15GB of VRAM. The CUDA version is 12.4, and the NVIDIA driver version is 550.54.15.

2.4. Monte Carlo Dropout Predictions with Uncertainty:

The Monte Carlo dropout method is used to estimate the uncertainty often associated with the prediction made by a neural network model. In a traditional neural network, in practice to avoid overfitting during the training of a neural network Dropout is usually used by randomly disabling neurons at each iteration.

If we allow dropout during the prediction phase, we can obtain an approximation of each prediction's uncertainty. Indeed, in contexts where errors must be considered critical the method of generating many different predictions for each input based on different subsets of neurons activated at each iteration allows us to estimate the variance of the predictions. Unlike traditional deterministic networks, Monte Carlo Dropout enables uncertainty estimation by performing multiple stochastic forward passes, leading to a probabilistic interpretation of predictions.

The Monte Carlo Dropout method is represented in Equation 6 with X the input data, θ is the model parameters,and Dropout Dr to obtain a random prediction $\hat{y}_i$. This method makes multiple stochastic predictions to estimate the uncertainty of the model. The model runs several simulations for every prediction by enabling dropout Dr (*training=True*) to get different predictions $\{\hat{y}_1, \hat{y}_2, ..., \hat{y}_{n_{sim}}\}$ for the same input. These predictions are then used to give the prediction mean, prediction standard deviation, and confidence interval governed by Equations 7,6,8 respectively. The prediction mean soothed the simulations to make the final prediction $\hat{y}_\mu$ that represents the best guess of the model. While the prediction standard deviation calculates $\sigma_{\hat{y}}$ quantifies the dispersion of predictions around the mean, a high dispersion (large $\sigma_{\hat{y}}$) means high uncertainty, while a low dispersion indicates a confident prediction. Finally, the confidence interval indicates that, for 95% of cases, the true value of the prediction is within the range $[\hat{y}_\mu - 2\sigma_{\hat{y}}, \hat{y}_\mu + 2\sigma_{\hat{y}}]$.

This information allows us to assess the model's reliability under various conditions. Figure 12 presents the Monte Carlo Dropout predictions with uncertainty.

$$\hat{y}_i = f(X; \theta, Dr) \tag{6}$$

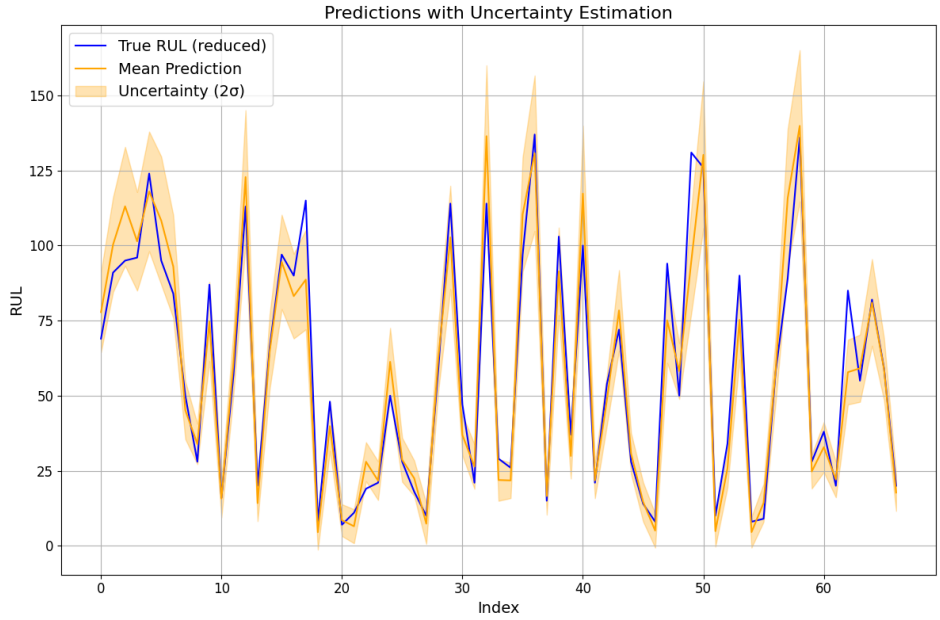
$$\hat{y}_\mu = \frac{1}{n_{\text{sim}}} \sum_{i=1}^{n_{\text{sim}}} \hat{y}_i \tag{7}$$

$$\sigma_{\hat{y}} = \sqrt{\frac{1}{n_{\text{sim}}} \sum_{i=1}^{n_{\text{sim}}} (\hat{y}_i - \hat{y}_\mu)^2} \tag{8}$$

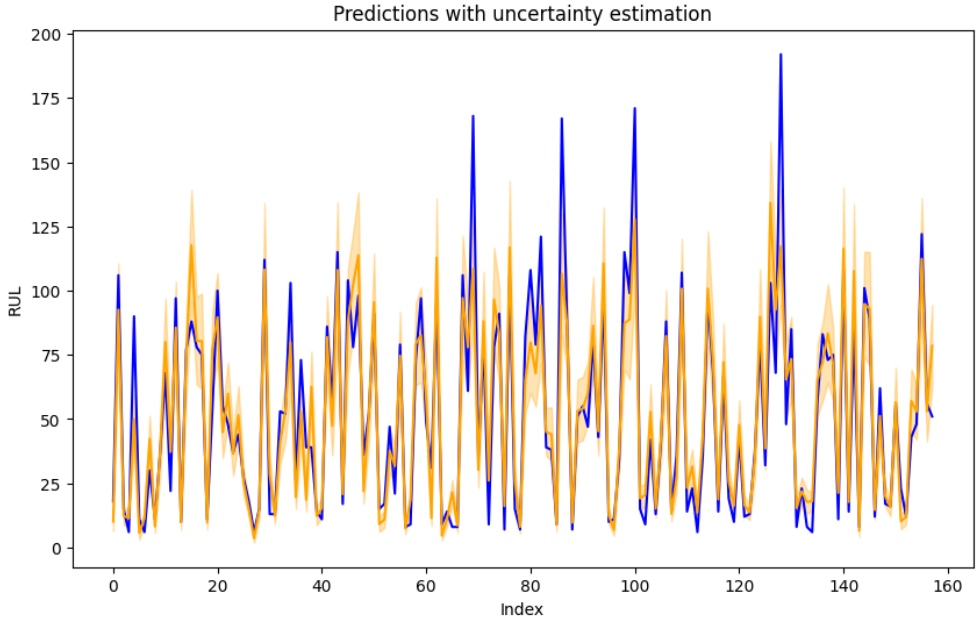
$$\hat{y}_\mu \pm 2\sigma_{\hat{y}} \tag{9}$$

3. RESULTS AND DISCUSSION

This section presents important results and discusses them in detail. It begins by analyzing the model's learning behavior, showcasing key illustrative results such as loss curves, uncertainty estimates, and error distributions. These are also compared with baseline models like TCN and LSTM to demonstrate that our contributions significantly enhance the model's



(a)



(b)

Figure 12. Monte Carlo Dropout Predictions with uncertainty: (a) FD001, (b) FD004

performance. Following this, a numerical evaluation is presented to highlight two critical aspects: uncertainty and prediction accuracy, as well as the generalization of the model. These results are provided in the form of tables and other visual representations.

3.1. Visual metics and comparisons

In Figure 13 , the loss function of the proposed model, TCN-LSTM, is showcased and compared to its submodels, TCN and LSTM, separately. The loss function of the TCN-LSTM model demonstrates smoother convergence with consistently lower loss values compared to the other models. While TCN and LSTM exhibit cases of rapid convergence over fewer epochs, this behavior may indicate a tendency toward overfitting or suboptimal generalization. In contrast, the TCN-LSTM's loss function highlights its ability to train effectively, mitigating overfitting and achieving better overall performance. This smooth convergence reflects the model's capacity to balance learning dynamics and optimize its representation of the data.

Additional results regarding the distributions of prediction error probabilities are showcased in Figure 14. When comparing these distributions, the proposed TCN-LSTM model exhibits lower variance in prediction errors compared to TCN and LSTM, with probability values slightly lower than those of LSTM. While LSTM demonstrates marginally higher probabilities at certain points, it is also characterized by higher variance. On the other hand, TCN shows the least favorable performance, with large variance and lower probability values overall. This indicates that TCN-LSTM achieves a more balanced performance by maintaining stable and consistent error distributions, which highlights its robustness and reliability in modeling tasks. The results emphasize that the integration of TCN and LSTM leverages the strengths of both submodels, while mitigating their individual weaknesses, leading to superior predictive capabilities.
Moreover, Figure 15 presents comparison of actual and predicted values and Figure 16 illustrates the results of uncertainty quantification using the Monte-Carlo method for some sensors, specifically T24 and W32. The figure showcases the spread of possible predictions, reflecting the uncertainty inherent in the model's estimates. Uncertainty, in this context, is a useful and important aspect. It allows the model to quantify how confident or uncertain it is in its predictions. High uncertainty regions highlight areas where the model might struggle or where further data might be needed, while low uncertainty regions suggest that the model is more confident in its predictions. The model in this case showcases a robust performances and provides a better understanding of the reliability of its predictions due to clear less uncertainty and less interval width of errors margins. This is particularly valuable in applications where decision-making depends on the predicted value and the confidence level associated with it.

Table 8. Metrics Results.

Dataset	Loss	MAE	RMSE	R² score
FD001	110.12	5.70	7.50	0.95
FD002	233.17	10.08	14.94	0.85
FD003	122.37	6.74	9.61	0.92
FD004	446.679	13.61	18.85	0.83

3.2. Numerical evaluation of model performance

After visually observing the model's behavior, this subsection is dedicated to presenting important numerical evaluations and comparisons, further strengthening the evidence for the proposed methodology. To this end, several key metrics are used to evaluate the model in this study(Naser & Alavi, 2023). These include regression metrics such as RMSE, MAE, MSE, and R^2 (as defined in equations 10, 11, 12, and 13), which will be presented for both training and validation phases. The obtained results of the proposed TCN-LSTM are compared to those of other state-of-the-art models presented in recent literature.

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

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

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

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{13}$$

First, Table 8 presents the results of the TCN-LSTM model across the four subsets of the CMAPSS dataset, each representing different operating conditions and failure modes. The table showcases individual results for the model. The proposed model demonstrates high prediction accuracy, particularly for FD001 and FD003, achieving RMSE values of 7.50 and 9.61, and R^2 scores of 0.95 and 0.92, respectively. These results suggest that the model effectively captures degradation patterns in systems with simpler operational conditions (FD001) as well as those with moderate complexities (FD003).

Table 9, on the other hand, compares the results obtained using the proposed TCN-LSTM model with those from recent literature. The results obtained by comparative approaches in the Table 9 are adopted from their original publications, as it is out of the scope of the current study to reimplement all reference approaches in the same environment. The TCN-LSTM model demonstrates a competitive RMSE of 7.86 on FD001, outperforming several state-of-the-art approaches, including (Tan & Teo, 2021), (Xin et al., 2023), and (Fan et al., 2023). Similarly, the model achieves RMSE values of

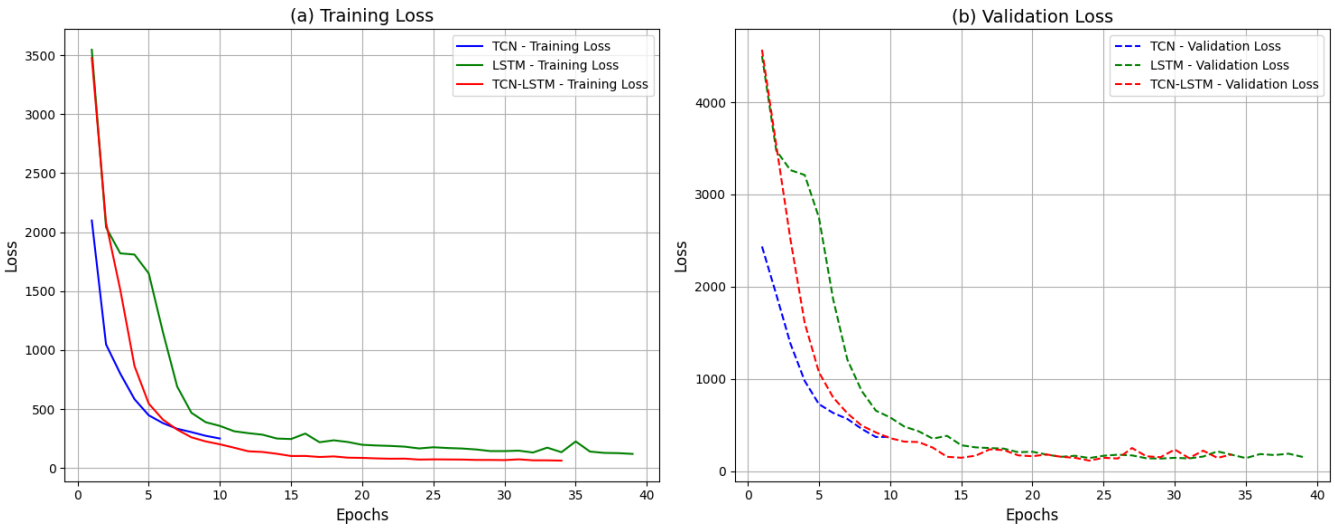


Figure 13. Loss function of studied models for both (a) training and (b)validation.

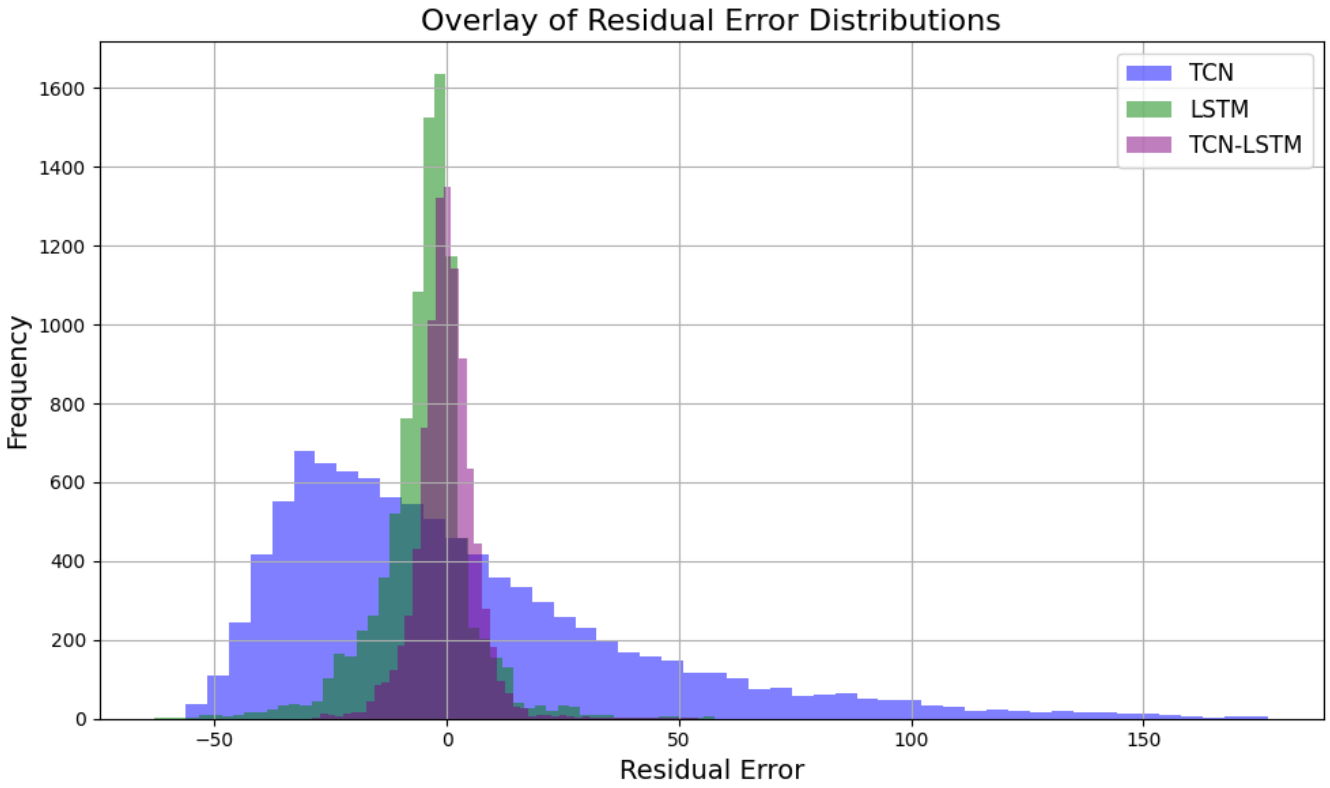
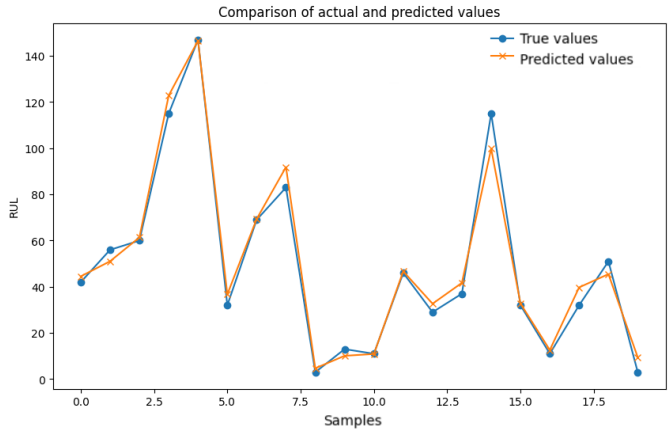
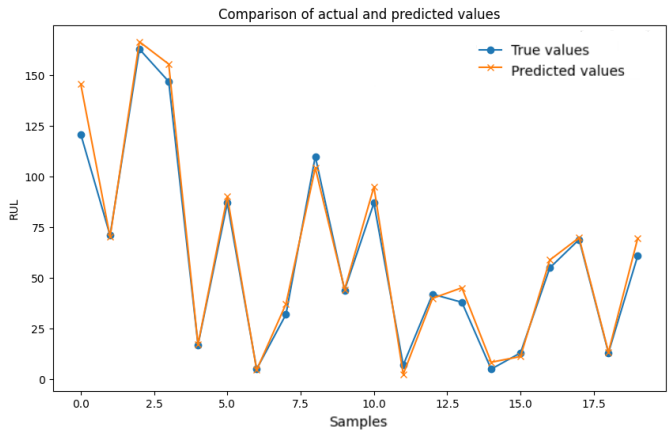


Figure 14. Overlay of residual error distributions.



(a)



(b)

Figure 15. Comparison of actual and predicted values,(a) FD001,(b) FD002

Table 9. Metrics Results: RMSE.

Model	FD001	FD002	FD003	FD004
(Deng & Zhou, 2024d)	15.97	14.45	13.90	16.63
(Chen et al., 2021)	14.90	15.19	16.71	19.74
(S. Wang et al., 2022)	11.47	16.15	10.74	18.90
(Fan et al., 2023)	10.98	16.12	11.14	18.15
(Tan & Teo, 2021)	10.60	14.55	11.71	17.23
(Xin et al., 2023)	10.35	15.82	11.34	17.35
(Ayodeji et al., 2021)	8.68	/	9.69	/
(Asif et al., 2022)	7.78	17.64	8.03	17.63
TCN-LSTM(2026)	**7.8578**	**14.94**	**9.61**	**18.85**

14.94 and 9.61 for FD002 and FD003, respectively, which are also competitive, demonstrating the model's robustness across varying operating conditions. Although Transformer-based models have recently demonstrated strong performance for RUL prediction, such as the BiLSTM-Autoencoder Transformer proposed by Fan et al. (Fan et al., 2023), the proposed TCN–LSTM model achieves superior predictive accuracy on the CMAPSS benchmark while maintaining a simpler architecture and lower computational complexity, making it particularly suitable for industrial PHM applications. Even if the model experiences a slight performance decline on FD004, its results still surpass those of several earlier models, as compared to (Chen, Chen, & Liu, 2021) and (S. Wang et al., 2022). The results of RMSE for FD004 are worse than the obtained for FD001 and FD004 , primarily due to the multiple simulation conditions(Yang, Zhao, Jiang, Sun, & Mei, 2019). FD004 is more complex than other CMAPSS subsets because it combines two failure modes (HPC degradation and Fan degradation), which increases the variability of the data. Additionally, the operating conditions in FD004 are more diverse and more challenging (setting six) , making it more difficult for the prediction models to learn. This complexity results in a higher RMSE than for FD001 and FD003.

Future optimization, alongside advanced preprocessing techniques, could address this performance gap.

3.2.1. Interpretation of the uncertainty results

After the Monte Carlo predictions, the same metrics used in subsection 3.2 were computed to assess the performance of the model.

The obtained results for FD001 show an RMSE of 9.52, and RMSE of 14.91 For FD002. This means that the model predicts fairly well: the average error is small, it has good penalization for large errors, and the model explains a large proportion of variance. These metrics, together with the estimated uncertainty, provide a general view of the performance and reliability of the predictions. It is relevant to indicate that when estimating the uncertainty for the predictions of the FD002 dataset, the performance of the TCN-LSTM model clearly improved compared to the other subsets.

The uncertainties are visualized in the graph in figure (12), showing the confidence interval around the mean predictions,

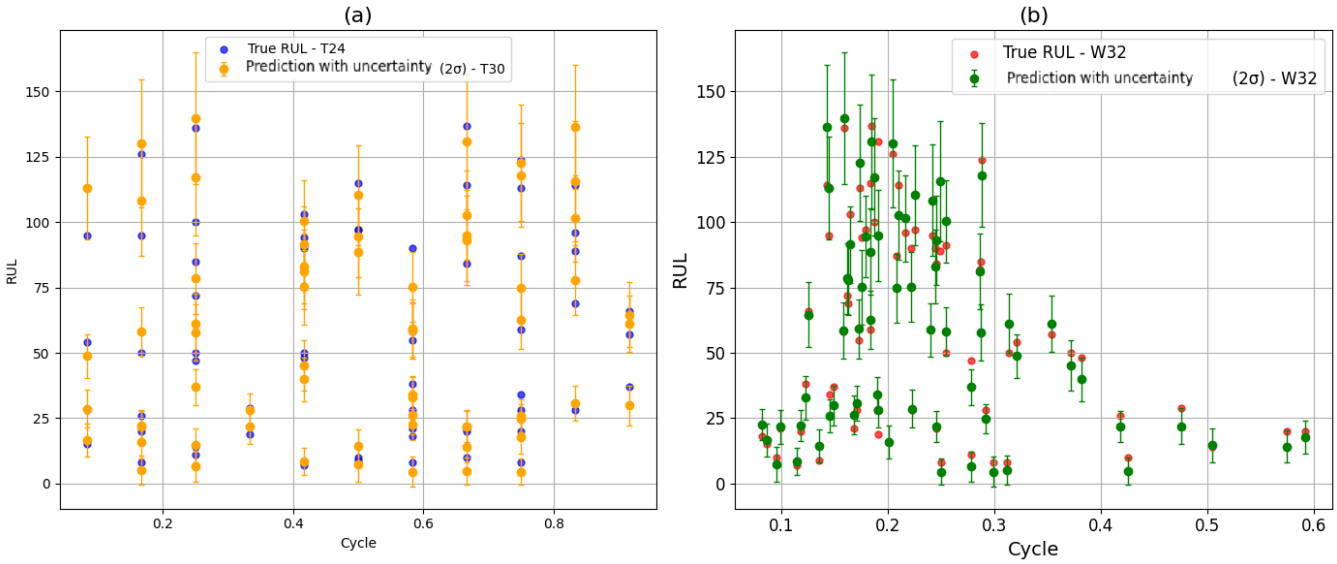


Figure 16. Predictions and uncertainty for (a) T24 and (b) W32.

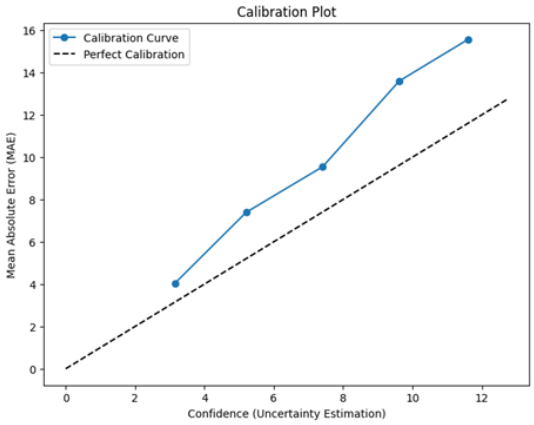


Figure 17. Expected Calibration Error for FD001.

calculated as ±2 times the standard deviation of the predictions for the T24 and W32 sensors.

As shown in Figure 17 the curve follows a smooth trend, suggesting a correspondence between uncertainty and mean error.

The deviation from the perfect calibration line is more systematic, meaning that the error is more predictable given the uncertainty.

4. CONCLUSION

In this work, we presented an innovative approach for performing anomaly detection and RUL prediction. The proposed hybrid approach provides a strong solution to time series anomaly detection and correction, combining the powers of unsupervised clustering in adaptive data segmentation and autoencoder in modeling data within each cluster. The method can significantly enhance the quality of the data quality compared to global approaches since it takes advantage of the heterogeneity among segmented data. This hybrid model has a distinct advantage by reducing sensitivity to anomalies and adapting to different operational modes. The proposed TCN-LSTM model has demonstrated state-of-the-art performance on several subdatasets (FD001-FD004) of the CMAPSS benchmark, achieving remarkable results on the FD001 and FD003 datasets. This model integrates Temporal Convolutional Networks for sequential feature extraction with LSTMs to capture long-term dependencies. In addition to the advantages described above, uncertainty quantification in the TCN-LSTM model makes the framework a trusted prediction by providing confidence intervals in conjunction with RUL predictions. This is an important feature for real-world applications where understanding the reliability of predictions is as key as the predictions themselves. Future work should explore advanced hyperparameter tuning methods, such as Bayesian optimization, and extend the framework to real-time streaming data for online RUL prediction.

REFERENCES

Alomari, Y., Baptista, M., & Ando, M. (2024). Integrating network theory and shap analysis for enhanced rul prediction in aeronautics. *PHM Society European Conference*, *8*, 15. doi: 10.36001/phme.2024.v8i1.4077

Arias Chao, M., Kulkarni, C., Goebel, K., & Fink, O. (2022). Fusing physics-based and deep learning models for prognostics. *Reliability Engineering & System Safety*, *217*, 107961. Retrieved from https://www.sciencedirect.com/science/

article/pii/S0951832021004725 doi: 10.1016/j.ress.2021.107961

Arias Chao, M., Kulkarni, C., Goebel, K., & Fink, O. (2021). Aircraft engine run-to-failure dataset under real flight conditions for prognostics and diagnostics. *Data*, *6*(1), 5. Retrieved from https://www.mdpi.com/2306-5729/6/1/5 doi: 10.3390/data6010005

Asif, O., Haider, S. A., Naqvi, S. R., Zaki, J. F. W., Kwak, K.-S., & Islam, S. M. R. (2022). A deep learning model for remaining useful life prediction of aircraft turbofan engine on cmapss dataset. *IEEE Access*, *10*, 95425–95440. doi: 10.1109/ACCESS.2022.3203406

Ayodeji, A., Wang, W., Su, J., Yuan, J., & Liu, X. (2021). An empirical evaluation of attention-based multi-head models for improved turbofan engine remaining useful life prediction. *arXiv preprint*. doi: 10.48550/arXiv.2109.01761

Berghout, T., & Benbouzid, M. (2022). A systematic guide for predicting remaining useful life with machine learning. *Electronics*, *11*(7), 1125. doi: 10.3390/electronics11071125

Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. *ACM Computing Surveys (CSUR)*, *41*(3), 1–58. doi: 10.1145/1541880.1541882

Chen, J., Chen, D., & Liu, G. (2021). Using temporal convolution network for remaining useful lifetime prediction. *Engineering Reports*, *3*(3), e12305. doi: 10.1002/eng2.12305

Deng, S., & Zhou, J. (2024a). Cnn for feature extraction in rul prediction. *International Journal of Computational Intelligence Systems*.

Deng, S., & Zhou, J. (2024b). Lstm-based rul prediction for aero-engines. *International Journal of Computational Intelligence Systems*.

Deng, S., & Zhou, J. (2024c). Prediction of remaining useful life of aero-engines based on cnn-lstm-attention. *International Journal of Computational Intelligence Systems*, *17*, 232. doi: 10.1007/s44196-024-00639-w

Deng, S., & Zhou, J. (2024d). Prediction of remaining useful life of aero-engines based on cnn-lstm-attention. *International Journal of Computational Intelligence Systems*, *17*, 232. doi: 10.1007/s44196-024-00639-w

Fan, Z., Li, W., & Chang, K.-C. (2023). A bidirectional long short-term memory autoencoder transformer for remaining useful life estimation. *Mathematics*, *11*(24), 4972. doi: 10.3390/math11244972

Genane, Y., & Aalah, A. (2023). An explainable artificial intelligence approach for remaining useful life prediction. *Aerospace*, *10*(5), 474. doi: 10.3390/aerospace10050474

Hawkins, D. M. (1980). *Identification of outliers*. Springer. doi: 10.1007/978-94-015-3994-4

Hong, C. W., Lee, C., Lee, K., Ko, M.-S., Kim, D. E., & Hur, K. (2020). Remaining useful life prognosis for turbofan engine using explainable deep neural networks with dimensionality reduction. *Sensors*, *20*(22), 6626. doi: 10.3390/s20226626

Hu, C., Cheng, F., Ma, L., & Li, B. (2022). State of charge estimation for lithium-ion batteries based on tcn-lstm neural networks. *Journal of the Electrochemical Society*, *169*, 030544. doi: 10.1149/1945-7111/ac5cf2

Hu, J., Jiang, X., Xu, H., & Zhang, K. (2025). Fault prediction of aircraft engine based on adaptive hybrid sampling and bilstm. *Scientific Reports*, *15*, 13726. doi: 10.1038/s41598-025-98756-9

Hu, T., Zhu, X., & Tian, M. (2025). Enhanced soc estimation method for lithium-ion batteries using bayesian-optimized tcn–lstm neural networks. *AIP Advances*, *15*. doi: 10.1063/5.0243811

Kong, X., Chen, Z., & Liu, W. (2025). Deep learning for time series forecasting: a survey. *Int. J. Mach. Learn. & Cyber.*. doi: 10.1007/s13042-025-02560-w

Lei, Y., Li, N., Guo, L., Yan, T., & Lin, J. (2018). Machinery health prognostics: A systematic review from data acquisition to rul prediction. *Mechanical Systems and Signal Processing*, *104*, 799–834. doi: 10.1016/j.ymssp.2017.11.016

Li, H., Zhang, Z., Li, T., & Si, X. (2024). A review on physics-informed data-driven remaining useful life prediction: Challenges and opportunities. *Mechanical Systems and Signal Processing*, *209*, 111120. Retrieved from https://www.sciencedirect.com/science/article/pii/S0888327024000189 doi: 10.1016/j.ymssp.2024.111120

Liu, L., Wang, L., & Yu, Z. (2021a). Deep convolutional networks for accurate rul prediction. *International Journal of Computational Intelligence Systems*.

Liu, L., Wang, L., & Yu, Z. (2021b). Lightgbm model for aircraft engine rul prediction. *International Journal of Computational Intelligence Systems*.

Liu, L., Wang, L., & Yu, Z. (2021c). Remaining useful life estimation of aircraft engines based on deep convolution neural network and lightgbm combination model. *International Journal of Computational Intelligence Systems*.

Liu, L., Wang, L., & Yu, Z. (2021d). Xgboost application in rul estimation of aero-engines. *International Journal of Computational Intelligence Systems*.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. *Berkeley Symp. on Math. Statist. and Prob.*, 281–297. doi: 10.48550/arXiv.2110.03585

Naser, M. Z., & Alavi, A. H. (2023). Error metrics and performance fitness indicators for artificial intelligence and machine learning in engineering and sciences. *Architecture, Structures and Construction*, *3*(4), 499–517. doi: 10.1007/s44150-021-00015-8

Sakurada, M., & Yairi, T. (2014). Anomaly detection using autoencoders with nonlinear dimensionality reduction. In *Proceedings of the mlsda 2014 2nd workshop on machine learning for sensory data analysis* (pp. 4–11). doi: 10.1145/2689746.2689747

Saxena, A., Goebel, K., Simon, D., & Eklund, N. (2008). Damage propagation modeling for aircraft engine run-to-failure simulation. In *Proceedings of the international conference on prognostics and health management.* doi: 10.1109/PHM.2008.4711414

Sharma, L., Singh, H., & Choudhary, M. P. (2025). Application of deep learning techniques for analysis and prediction of particulate matter at kota city, india. *EQA - International Journal of Environmental Quality*, *66*, 107–115. doi: 10.6092/issn.2281-4485/20687

Tan, W. M., & Teo, T. H. (2021). Remaining useful life prediction using temporal convolution with attention. *AI*, *2*(1), 48–70. Retrieved from https://www.mdpi.com/2673-2688/2/1/5 doi: 10.3390/ai2010005

Wang, S., Ji, B., Wang, W., Ma, J., & Chen, H.-B. (2022). Remaining useful life prediction of aero-engine based on deep convolutional lstm network. In *6th international conference on system reliability and safety (icsrs)* (pp. 494–499). doi: 10.1109/ICSRS56243.2022.10067647

Wang, X., Li, Y., Xu, Y., Liu, X., Zheng, T., & Zheng, B. (2023). Remaining useful life prediction for aero-engines using a time-enhanced multi-head self-attention model. *Aerospace*. doi: 10.3390/aerospace10010080

Wang, X., Li, Y., Xu, Y., Liu, X., Zheng, T., & Zheng, B. (2024). Improved feature selection for rul prediction using temporal attention mechanisms. *Aerospace*.

Wang, X., Zhao, Y., & Pourpanah, F. (2020). Recent advances in deep learning. *Int. J. Mach. Learn. & Cyber.*, *11*, 747–750. doi: 10.1007/s13042-020-01096-5

Xin, f. n. m., Wang, Y., Li, Y., Xu, X., Liu, T., & Zheng, B. (2023). Remaining useful life prediction for aero-engines using a time-enhanced multi-head self-attention model. *Aerospace*, *10*. doi: 10.3390/aerospace10010080

Xu, H., Chen, W., Zhao, N., Li, Z., Bu, J., Li, Z., ... Deng, S. (2019). Deep learning-based anomaly detection in multivariate time series data. In *Proceedings of the 28th international joint conference on artificial intelligence (ijcai)* (pp. 2325–2331). doi: 10.24963/ijcai.2019/322

Xu, R., & Wunsch, D. (2005). Survey of clustering algorithms. *IEEE Transactions on Neural Networks*.

Yang, H., Zhao, F., Jiang, G., Sun, Z., & Mei, X. (2019). A novel deep learning approach for machinery prognostics based on time windows. *Applied Sciences*, *9*(22), 4813. Retrieved from https://www.mdpi.com/2076-3417/9/22/4813 doi: 10.3390/app9224813

BIOGRAPHIES

N.Ouafek is lecturer in computer science at Higher Normal School-Constantine, where she has been a lecturer in computer science since 2009. She earned her PhD in 2019 from the University of Constantine 2 in machine learning for predictive maintenance and anomaly detection. Her research focus on the use of cutting-edge deep learning techniques such as LSTM networks, variational autoencoders (VAE) and attention mechanisms in complex industrial environments AND Images. She is also interested in science popularization as well as student project guidance combining artificial intelligence with social issues.