Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Temporal Degradation Pattern Learning for Aircraft Engine Remaining Useful Life Prediction

Aditi Gawde

aditiarjun.gawde45@nmims.in

Narsee Monjee Institute of Management Studies

Avinash More

Narsee Monjee Institute of Management Studies

Research Article

Keywords: Remaining useful life, Temporal embedding, Temporal convolutional neural network, Aircraft engine degradation, Prognostics and health management

Posted Date: October 1st, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11017626/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Temporal Degradation Pattern Learning for Aircraft Engine Remaining Useful Life Prediction

Aditi Gawde[1*] and Avinash More[2*†]

[1*]Department of Electronics and Telecommunications, Mukesh Patel School of Technology, Management and Engineering, Mumbai, 400056, Maharashtra, India.

*Corresponding author(s). E-mail(s): aditiarjun.gawde45@nmims.in; Avinash.More@nmims.edu;
†This author contributed equally to this work.

Abstract

Predicting the Remaining Useful Life (RUL) of aircraft engines is vital for reliable operation and safe, efficient maintenance planning. While deep learning methods have made significant progress in RUL prediction, many convolutional models struggle to capture evolving engine degradation patterns under varying operating conditions. In addition, fixed sinusoidal positional encodings lack the flexibility to represent complex temporal degradation patterns. This study proposes a Temporal Embedding-based Temporal Convolutional Neural Network (TE-TCNN), in which a learnable Temporal Embedding (TE) module is jointly trained with the network to adapt the temporal representation to aircraft-engine degradation patterns. The proposed framework is evaluated using fixed kernel sizes of 3, 5, and 7 to investigate the effect of the temporal receptive field on prediction performance. To support robust degradation representation, a Health Index (HI) is developed using Principal Component Analysis (PCA) with trajectory smoothing and monotonicity constraints. The engineering application is aircraft-engine RUL prediction using all four NASA Commercial Modular Aero-Propulsion System Simulation (C-MAPSS) benchmark datasets (FD001-FD004). The proposed TE-TCNN demonstrates its ability to learn aircraft-engine degradation patterns efficiently, supporting intelligent predictive maintenance of aircraft engines.

Keywords: Remaining useful life, Temporal embedding, Temporal convolutional neural network, Aircraft engine degradation, Prognostics and health management.

1 Introduction

Aircraft engines work in tough and ever-changing conditions during their whole life, which makes it hard to plan reliable maintenance in today's aviation.The ability to predict how much longer an engine can work before it breaks down helps schedule maintenance before serious problems happen, making flying safer, improving how well systems work, and lowering the cost of repairs.The increasing use of sensors in aircraft engines has led to the collection of large amounts of data from different engine components. Such data provides opportunities for data-driven approaches in Predictive Maintenance and Health Management. In contrast to approaches that rely primarily on predefined physical models or failure mechanisms, data-driven methods learn degradation patterns directly from historical sensor measurements and operating conditions.

Despite recent progress, Remaining Useful Life (RUL) prediction remains challenging because engine degradation does not follow a simple pattern and can vary with operating conditions. Sensor measurements may contain uncertainty and can change with the way an engine is operated, making it difficult to distinguish degradation-related changes from normal operating variations. These challenges are particularly evident in the C-MAPSS benchmark datasets, with FD004 being one of the more complex subsets because it combines multiple operating conditions and fault modes. Therefore, developing models that can represent temporal degradation behaviour while maintaining consistent prediction performance across different conditions remains an important research problem.

Deep learning methods have been widely investigated for RUL prediction because they can learn features directly from multivariate sensor data. Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, Convolutional Neural Networks (CNNs), and Transformer-based models have all been applied to this problem. CNNs are attractive for this application because of their relatively simple structure and their ability to extract local patterns from sequential data. However, conventional CNNs do not explicitly represent the temporal position of observations, which can limit their ability to distinguish degradation patterns occurring at different stages of an engine's operating life. Fixed positional representations also provide limited adaptability when degradation behaviour changes across operating conditions. Transformer-based approaches can model relationships between observations over longer ranges, but their attention mechanisms generally require greater computational and memory resources, which can limit their suitability for lightweight deployment.

To address these limitations, this study proposes a Temporal Embedding-based Temporal Convolutional Neural Network (TE-TCNN) for aircraft engine RUL prediction. The proposed approach introduces a learnable temporal embedding into the input sequence before convolutional feature extraction. This allows the temporal representation to be optimized jointly with the convolutional network while retaining the relatively lightweight structure of a CNN. The proposed framework combines a lightweight temporal convolutional network with a learnable temporal embedding module that is optimized jointly during training. Instead of using fixed sinusoidal positional encoding, the learnable embedding adapts to the degradation characteristics present in the training data, allowing the network to capture temporal progression

more effectively while preserving the computational efficiency of convolutional architectures. To further examine the robustness of the proposed framework, a kernel sensitivity analysis is performed using convolutional kernel sizes of 3, 5, and 7 to investigate the influence of receptive field size on feature extraction and prediction accuracy. Furthermore, a Health Index (HI) is constructed using Principal Component Analysis (PCA) and refined through smoothing and monotonicity constraints to generate stable and interpretable degradation trajectories for monitoring engine health.

The major contributions of this work are summarized as follows:

- Proposed a **Temporal Embedding-based Temporal Convolutional Neural Network (TE-TCNN)** for accurate Remaining Useful Life prediction of aircraft engines by integrating adaptive temporal representation learning with a lightweight convolutional architecture.
- Introduced a **learnable temporal embedding module** that adaptively captures degradation progression during training, allowing the temporal representation to be optimized jointly with the network parameters.
- Developed a **PCA-based Health Index (HI)** framework with smoothing and monotonicity constraints to generate stable and interpretable degradation trajectories for engine health assessment.
- Performed a comprehensive **kernel sensitivity analysis** using different convolutional kernel sizes to evaluate the influence of receptive field size on feature extraction and prediction performance.
- Evaluated the proposed **TE-TCNN** on all four NASA C-MAPSS benchmark datasets (FD001–FD004), demonstrating competitive performance in terms of both RMSE and NASA Score, particularly on the challenging FD004 dataset.

The proposed framework is evaluated using all four NASA C-MAPSS benchmark datasets with the standard Root Mean Square Error (RMSE) and NASA Scoring Function as evaluation metrics. Experimental results show that the proposed model consistently provides reliable RUL predictions across different operating conditions and fault modes while maintaining a lightweight architecture that is well suited for practical predictive maintenance applications.

The remainder of this paper is organized as follows. Section 2 reviews existing research on data-driven RUL prediction methods. Section 3 describes the proposed methodology, including data preprocessing, temporal embedding, and the TE-TCNN architecture. Section 4 presents the experimental setup and discusses the obtained results. Section 5 provides a detailed discussion of the findings, and Section 6 concludes the paper.

2 Related Works

Remaining Useful Life (RUL) prediction has emerged as a key research problem in Prognostics and Health Management (PHM), particularly following the release of the NASA C-MAPSS turbofan engine degradation dataset by Saxena and Goebel [1]. The dataset provides multivariate sensor measurements collected from turbofan engines under different operating conditions and degradation patterns and has subsequently

become a widely adopted benchmark for evaluating data-driven prognostic methods. Early RUL prediction studies primarily relied on statistical and stochastic degradation models to estimate system health and predict remaining operational life. Lei et al. [2] proposed a stochastic process-based framework for machine RUL prediction, while Zhang et al. [3] investigated a multi-objective deep belief network ensemble for prognostic estimation. Although these approaches provided important foundations for data-driven prognostics, their ability to capture highly nonlinear relationships within high-dimensional sensor measurements can be limited, particularly when degradation behaviour varies under complex operating conditions.

The emergence of deep learning significantly advanced RUL prediction by enabling automatic feature extraction and representation learning from multivariate sensor data. Recurrent Neural Networks (RNNs) were among the earliest deep learning architectures applied to prognostic tasks because of their ability to process sequential information. Heimes [4] demonstrated the feasibility of RNNs for aircraft-engine RUL estimation. However, conventional RNNs can experience difficulties in learning long-term dependencies because of vanishing and exploding gradients. Long Short-Term Memory (LSTM) networks were subsequently adopted to address these limitations through gated memory mechanisms. Zheng et al. [5] demonstrated the effectiveness of LSTM networks for RUL estimation by learning temporal dependencies from degradation sequences. Gated Recurrent Unit (GRU) architectures further simplified recurrent processing while maintaining competitive temporal modelling capability [6]. Despite their effectiveness in modelling sequential degradation information, recurrent architectures process sequence elements recursively, which limits parallel computation and can increase training time and computational cost.

Convolutional Neural Networks (CNNs) have consequently emerged as an efficient alternative because of their ability to automatically extract degradation-related features through convolutional operations while enabling highly parallel computation. Babu et al. [7] introduced a deep CNN-based regression approach for RUL estimation, demonstrating the effectiveness of convolutional feature extraction for prognostic prediction. Li et al. [8] further investigated deep convolutional neural networks for RUL estimation and demonstrated their capability to learn useful degradation representations from sensor measurements. Later research focused on multi-scale CNN structures to identify degradation features at various levels [9], and ensemble CNN models were created to boost prediction reliability [10].While CNN-based techniques are effective at extracting features, traditional convolutional operations mainly detect local patterns using their receptive fields and do not clearly represent the time-based location or development of degradation over a sequence.

To combine the advantages of convolutional and recurrent models, hybrid deep learning approaches have also been studied. CNN-RNN and CNN-LSTM models typically use convolutional layers to detect local degradation patterns and recurrent units to represent how these features evolve over time. These models can enhance prediction accuracy by combining local feature extraction with the modeling of temporal relationships. However, the use of recurrent components requires sequential processing, which increases model complexity, training time, and computational requirements.

These limitations have motivated the development of convolutional models that can capture temporal dependencies without relying on recurrent mechanisms.

Temporal Convolutional Networks (TCNs) have therefore emerged as an effective alternative for sequential modelling. Ren et al. [11] proposed a Temporal Convolutional Memory Network for RUL estimation of industrial machinery by combining temporal convolution with memory mechanisms. Compared with recurrent architectures, temporal convolution can provide larger receptive fields, stable gradient propagation, and efficient parallel computation while retaining temporal information. Nevertheless, temporal information in conventional convolutional architectures is primarily determined by the convolutional receptive field, and the model does not necessarily possess an explicit mechanism for adaptively representing the temporal position of observations within a degradation sequence.

Attention mechanisms and Transformer-based architectures have subsequently provided more advanced approaches for modelling long-range temporal dependencies. Vaswani et al. [12] introduced the Transformer architecture, demonstrating that self-attention can capture relationships within sequential data without relying on recurrent computation. Zhou et al. [13] subsequently proposed Informer to improve the efficiency of attention-based modelling for long-sequence time-series data. Liu et al. [14] investigated a Fast Temporal Transformer-GRU framework for RUL prediction, while Jin et al. [15] proposed PE-Net, which incorporates positional encoding into a convolutional architecture to preserve temporal ordering during feature extraction. These studies highlight the importance of effective temporal representation and long-range dependency modelling for degradation sequences. However, Transformer-based approaches can introduce increased architectural and computational complexity, particularly when modelling long sequences. Moreover, fixed positional representations may provide limited adaptability to degradation patterns that vary across operating conditions and degradation behaviours. These limitations motivate the exploration of more adaptive temporal representations that can be integrated with computationally efficient convolutional architectures.

Attention-based feature enhancement has also been explored to improve the representation of informative sensor measurements. Woo et al. [16] proposed the Convolutional Block Attention Module (CBAM), which adaptively refines channel and spatial features and provides a mechanism for emphasizing informative features within high-dimensional representations. Knowledge-distillation approaches have similarly been investigated to reduce the computational requirements of deep prognostic models. Zhang et al. [17] introduced KDNet, which transfers knowledge from a more complex teacher model to a simpler student network. The approach aims to retain the predictive capability of the teacher while reducing the complexity of the resulting model. Such model compression strategies are relevant to predictive maintenance applications where computational and memory resources may be limited.

Other studies have investigated data augmentation and physics-guided learning to improve the reliability of RUL prediction under limited or variable data conditions. Generative adversarial networks (GANs), for example, have been used to generate additional degradation trajectories and increase the diversity of the training data.

Raissi et al. [18] introduced Physics-Informed Neural Networks (PINNs), which incorporate physical constraints into the neural network training process. Physics-guided approaches can provide useful physical consistency when purely data-driven models are insufficient. However, the use of additional data-generation mechanisms, physical constraints, or separate training stages can also increase the overall complexity of the prediction framework.

Recent studies have also focused on practical challenges such as varying operating conditions, limited degradation information, incomplete sensor observations, and complex temporal dependencies. Jiang et al. [19] investigated RUL prediction across different operating conditions using domain-adaptation-based feature transfer, highlighting the need for degradation features that remain useful across different environments. This consideration is particularly relevant to the C-MAPSS benchmark, whose subsets differ in their operating conditions and fault characteristics.

Wang et al. [20] proposed a hybrid framework that combines a large-kernel CNN with a Markov feature framework for RUL prediction. Their results demonstrate the continued use of convolutional architectures for extracting degradation-related features while also considering a broader temporal context. The use of larger convolutional kernels provides one way of increasing the receptive field without relying on recurrent processing.

The interaction between multiple sensor channels has also received increasing attention in recent RUL prediction studies. Zhang et al. [21] proposed a multi-scale cross-channel attention network for RUL prediction with variable sensors, emphasizing the importance of modelling interactions between sensor channels while extracting features at multiple scales. Graph-based architectures provide another approach for explicitly representing relationships among heterogeneous sensor variables. Wen et al. [22] proposed a temporal and heterogeneous graph neural network for RUL prediction, combining temporal information with structured relationships among sensor variables. Although such approaches can provide richer representations of multivariate degradation behaviour, their additional architectural components may increase model complexity.

Transformer-based RUL prediction has also continued to develop through more sophisticated spatio-temporal representations. Kim et al. [23] proposed an enhanced spatio-temporal Koopman dual-branch Transformer for turbofan-engine RUL prediction, while Chen [24] proposed a Transformer-based deep-learning model enhanced by position-sensitive attention and gated hierarchical LSTM mechanisms for aero-engine RUL prediction. These studies further demonstrate the importance of temporal position and long-range dependency modelling for aircraft-engine degradation. However, the combination of Transformer, attention, and recurrent components can increase architectural complexity and computational requirements compared with lightweight convolutional approaches.

Another recent research direction focuses on reducing the dependence on large quantities of labelled run-to-failure data. Sun et al. [25] investigated few-shot RUL prediction for engines using a CNN-GRU architecture, addressing situations in which only limited degradation trajectories are available. Liu et al. [26] investigated self-supervised learning for RUL prediction using simple triplet networks, demonstrating

the potential to learn degradation representations with reduced dependence on labelled samples. These approaches address the data-efficiency problem, although additional learning stages or representation-learning mechanisms can increase the complexity of the training framework.

Uncertainty estimation and robustness have also become increasingly important in recent RUL research. Niu et al. [27] proposed a hybrid Gaussian process regression framework with temporal feature extraction for partially interpretable RUL interval prediction in aero-engine prognostics, highlighting the importance of uncertainty-aware prediction. Ness [28] investigated RUL prediction against adversarial attacks using an active-learning approach, emphasizing the importance of model robustness when sensor measurements are perturbed or unreliable. These studies extend RUL prediction beyond conventional point-estimation accuracy toward uncertainty quantification and robustness in practical operating environments.

Recent review studies further demonstrate the rapid development of artificial-intelligence-based approaches for RUL prediction. Wu et al. [29] provided a survey of deep-learning-based RUL prediction methods, while Li et al. [30] presented a comprehensive review of artificial-intelligence-based algorithms for predicting the RUL of equipment. Wang et al. [31] specifically reviewed graph neural network approaches for RUL prediction, covering their methodologies, evaluation strategies, and future research directions. Chen and Liu [32] further investigated multidimensional industrial signal processing and efficient transfer learning for RUL prediction using large language models. These studies collectively demonstrate the transition of RUL prediction from conventional statistical and recurrent approaches toward CNN, TCN, attention, Transformer, graph-based, self-supervised, domain-adaptation, and transfer-learning architectures.

Despite these significant advances, existing approaches continue to exhibit a trade-off between predictive performance, temporal representation capability, and computational efficiency. Recurrent architectures can effectively model sequential dependencies but rely on sequential computation, whereas conventional CNNs provide efficient parallel feature extraction but generally lack explicit adaptive mechanisms for representing temporal degradation progression. TCN-based approaches improve temporal modelling while retaining convolutional efficiency, but their temporal representation is largely determined by the convolutional receptive field. Transformer- and attention-based architectures provide powerful long-range dependency modelling but generally introduce greater computational and architectural requirements. Similarly, domain-adaptation, graph-based, self-supervised, and uncertainty-aware approaches address important practical challenges but introduce additional modelling components [19, 22, 26, 27]. Recent large-kernel CNN research further demonstrates the importance of receptive-field design for capturing degradation characteristics efficiently [20].

Motivated by these limitations, this work proposes a Temporal Embedding-based Temporal Convolutional Neural Network (TE-TCNN), which integrates a learnable temporal embedding mechanism within a lightweight convolutional framework. Unlike fixed positional representations, the proposed temporal embedding is jointly optimized with the network, allowing temporal information to be learned adaptively

from degradation data. The proposed approach therefore combines the parallel computational advantages of convolutional feature extraction with explicit temporal representation learning, aiming to achieve an effective balance between predictive performance, temporal modelling capability, and computational efficiency for practical RUL prediction.

3 Methodology

The workflow in Fig. 1 begins with the C-MAPSS dataset, which comprises multivariate time-series sensor measurements collected from aircraft engines operating under different degradation conditions.

Initially, data preprocessing is performed by removing non-informative sensors and computing the Remaining Useful Life (RUL) labels for each engine. To account for varying operating conditions, clustering-based normalization is employed to standardize the sensor measurements within similar operating regimes. Subsequently, a sliding window strategy is applied to transform the continuous time-series data into fixed-length input sequences suitable for model training.

The resulting sequences are then fed into the proposed Temporal Embedding-based Temporal Convoultion Neural Network (TE-TCNN). A learnable temporal embedding module is first incorporated to adaptively encode temporal degradation progression before convolutional feature extraction. The extracted temporal features are subsequently processed by the lightweight convolutional network to estimate the corresponding RUL values. Finally, the predictive performance of the proposed model is evaluated using the Root Mean Square Error (RMSE) and NASA Scoring Function metrics.

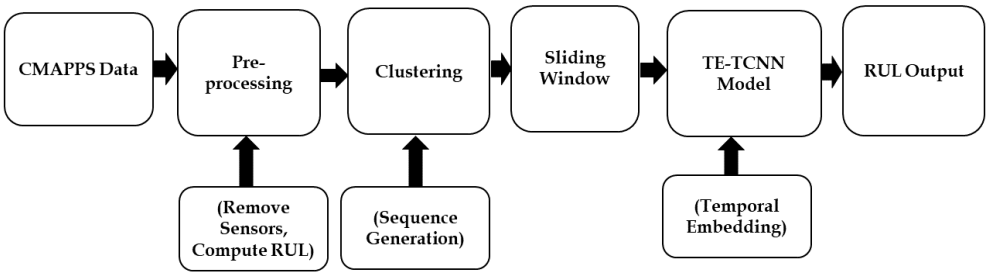


Fig. 1: Overall framework of the proposed method

3.1 Dataset Description

The experiments are carried out on all four subsets of the CMAPSS dataset. These include both simple and complex cases, with operating conditions of 1 and 6,and fault modes of 1 and 2. This helps in checking how well the model performs under different situations. The dataset is used with its original train and test split.

3.2 Data Pre-processing

Table 1: Dataset Description

Parameter	Value
Dataset	C-MAPSS dataset
Subsets	FD001–FD004
Operating Conditions	1 and 6
Fault Modes	1 and 2

Before training the proposed model, the C-MAPSS dataset undergoes a series of pre-processing steps to improve data quality and generate suitable input features for Remaining Useful Life (RUL) prediction. The preprocessing procedure consists of the following stages:

1) Sensor Selection: The original C-MAPSS dataset contains operational settings and multiple sensor measurements recorded during each engine cycle. However, not all sensors contribute significantly to degradation modelling. Sensors exhibiting little variation or containing redundant information are removed to reduce noise and computational complexity. Consequently, only the informative sensor measurements along with the operational settings are retained for subsequent analysis.

2) Remaining Useful Life (RUL) Computation: For each engine, the Remaining Useful Life is calculated as the difference between the final operating cycle and the current cycle. This computed RUL serves as the target value for supervised learning and represents the number of cycles remaining before engine failure.

3) RUL Capping: The calculated RUL values are capped at a maximum threshold of 125 cycles. Since engines operating in the early stages of their lifetime exhibit large RUL values with minimal degradation, retaining these values may dominate the training process. Limiting the maximum RUL produces a piecewise target representation, allowing the model to focus more effectively on the progressive degradation stage where accurate prediction is most critical.

The processed dataset is subsequently used for operating-condition handling, sliding-window construction, and training of the proposed TE-TCNN model.

3.3 Sliding Window Construction

The CMAPSS dataset includes engine operation data with different lengths of time-series sequences, which complicates the direct use of the data for model training.To deal with this issue, a sliding window method is used to convert the raw time-series data into sequences of fixed length.

For each engine, a window of a specified length W is moved across the sensor readings. In this study, a window size of 30 is applied to the FD001, FD002, and FD003 datasets, while a larger window size of 60 is used for FD004 to better represent its more complex degradation process.

At each step visualized in Fig. 2, the sensor readings inside the window are put together to make one input sample. The Remaining Useful Life (RUL) for this sample comes from the last cycle in the window, because it shows the most recent health condition of the engine. While training, the window moves one cycle at a time, which creates overlapping sequences. This helps make more training examples and lets the

Algorithm 1 TE-TCNN for RUL Prediction

Require: Training dataset D_{train}, testing dataset D_{test}, sliding window length W, maximum RUL threshold $RUL_{max} = 125$, learning rate η, batch size B, and number of training epochs E

Ensure: Predicted RUL, RMSE, and NASA Score

Step 1: Data Preprocessing
Remove non-informative sensors and compute clipped RUL labels
Cluster the operating conditions (op_1, op_2, op_3) using K-means
Perform cluster-wise standard normalization on the sensor features
Generate fixed-length input sequences using the sliding window approach
Step 2: Model Construction
Initialize a learnable temporal embedding matrix E
For each input sequence X, compute $X \leftarrow X + E$ to obtain the temporally embedded sequence
Pass the embedded sequence through:
    1-D Convolution Layer (Kernel = 5, Filters = 64)
    Batch Normalization + ReLU
    1-D Convolution Layer (Kernel = 5, Filters = 128)
    Batch Normalization + ReLU
Select the feature vector corresponding to the last time step
Apply Dropout
Predict RUL using the Fully Connected layer
Step 3: Model Training
for epoch = 1 to E **do**
    Divide the training data into mini-batches
    Compute the predicted RUL
    Compute the Mean Squared Error (MSE) loss
    Update the model parameters using the Adam optimizer
    Apply gradient clipping for stable training
end for
Step 4: Prediction
Predict the RUL for all test engine sequences using the trained model
Step 5: Performance Evaluation
Compute the Root Mean Square Error (RMSE)
Compute the NASA Scoring Function
return Predicted RUL, RMSE, and NASA Score

model understand how the engines condition changes over time. For testing, only the most recent window for each engine is used to predict how much life is left.

If a sequence is shorter than the window size, the first sensor values are repeated to fill in the missing data. This makes sure all input sequences are the same length. This method turns data of different lengths into uniform sequences, helping the model learn patterns in how the engine wears down over time. Whereas, the windowing is done separately for each engine to keep the time sequence continuous. All overlapping

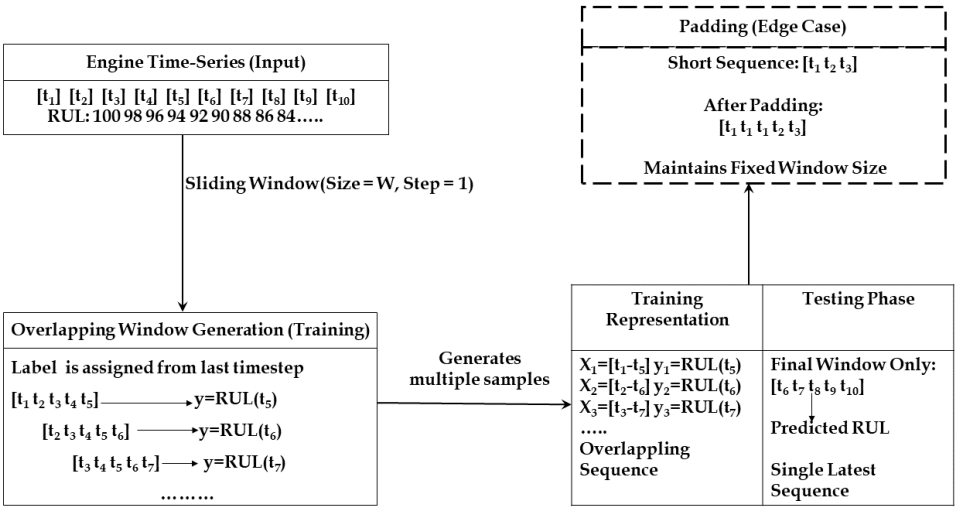


Fig. 2: Sliding window approach for sequence construction

windows are used during training, but only the final window is used during testing for prediction.

3.4 Condition Handling

The C-MAPSS dataset contains data from engines operating under different conditions. Because of these changes in operating conditions, the sensor readings can vary even when the actual health of the engine is similar. If these variations are not considered, the model may learn differences caused by the operating condition instead of the degradation of the engine.

To handle this variation, the three operational settings (op1, op2, and op3) are first grouped using K-means clustering. In this study, six clusters are used to represent groups of similar operating conditions. This gives a way to separate the observations according to the conditions under which the engine was operating.

After the clusters are obtained, the input features are normalized separately within each cluster using StandardScaler. In this way, the scaling of the data is based on the corresponding operating condition rather than on all observations together. This makes the input data more consistent across different operating conditions and helps the model focus on changes associated with engine degradation.

3.5 Temporal Embedding-Temporal Convoultion Neural Network (TE-TCNN) Architecture

Instead of conventional machine learning methods that process individual observations, the proposed model works with multivariate time-series sequences to learn changes in engine degradation. The TE-TCNN combines a learnable temporal embedding with a lightweight convolutional network. The temporal embedding provides information about the position of each observation in the sequence, while the convolutional layers extract degradation-related features from neighbouring time steps.

The input consists of fixed-length multivariate sequences generated using a sliding-window approach. Each sequence contains sensor measurements from consecutive engine cycles, allowing the model to use information from a short history of engine operation. Before the convolutional layers, a learnable temporal embedding matrix is added to the input sequence. Unlike fixed sinusoidal positional encodings, the values of this embedding are updated during model training. This allows the temporal representation to be learned along with the other model parameters while preserving the original order of the observations.

The embedded sequences are then passed through two one-dimensional convolutional layers. The first convolutional layer contains 64 filters with a kernel size of 5, followed by a second layer containing 128 filters with the same kernel size. These layers operate along the temporal dimension and extract features from neighbouring observations in the input sequence. Batch Normalization is applied after each convolutional layer, followed by the ReLU activation function. The two convolutional layers progressively transform the input sequence into higher-level feature representations. The feature vector corresponding to the final time step is then used for RUL prediction through the subsequent dropout and fully connected layers.Rather than aggregating information across the entire sequence, the proposed architecture selects the feature vector corresponding to the final time step of the last convolutional layer. This feature vector represents the latest health condition of the engine and therefore contains the most relevant degradation information for Remaining Useful Life (RUL) estimation.

To improve the generalization capability of the proposed model and mitigate overfitting, a dropout layer with a dropout probability of 0.2 is applied before the final prediction stage. The resulting feature representation is then passed through a fully connected regression layer, which estimates the Remaining Useful Life for each input sequence.

During the training phase, the learnable temporal embedding and network parameters are jointly optimized using the Adam optimizer by minimizing the Mean Squared Error (MSE) loss function. To ensure stable optimization and prevent exploding gradients, gradient clipping is employed throughout the training process. Once training is complete, the proposed Temporal Embedding-based Temporal Convolutional Neural Network (TE-TCNN) is used to predict the Remaining Useful Life of unseen engine sequences. The predictive performance of the model is subsequently evaluated using two widely adopted prognostic metrics: the Root Mean Square Error (RMSE) and the NASA Scoring Function.

3.6 Health Index Construction

Represented in Fig. 3 is the degradation behaviour of aircraft engines, a Health Index (HI) is constructed from the multivariate sensor measurements using Principal Component Analysis (PCA). Initially, the raw sensor data is preprocessed by removing non-informative sensors, followed by standard normalization to eliminate variations in feature magnitude and ensure that all sensor measurements contribute equally to the analysis.

PCA is then applied to the normalized sensor data to reduce its dimensionality while retaining the key information related to engine degradation.The health condition of each engine is represented using a Health Index (HI) obtained from the sensor data.

First, PCA is applied to the selected features, and the first principal component is used to summarize the main variation in the data. The resulting values are then normalized between 0 and 1. Since the direction of the principal component can vary, the values are inverted so that a higher HI represents a healthier engine condition and a lower HI represents a more degraded condition.

The normalized HI is then smoothed using a Savitzky–Golay filter with a window length of 11 and a polynomial order of 2. This reduces small fluctuations in the HI values while retaining the overall shape of the degradation curve. After smoothing, a cumulative minimum operation is applied so that the HI does not increase at later cycles. This gives a gradually non-increasing health trajectory over the engine life.

The resulting HI trajectories are used to examine the degradation behaviour of the engines in the four C-MAPSS subsets. The plots provide a visual representation of how the estimated engine condition changes from the earlier cycles toward the later stages of operation.

$$HI = Z \cdot p_1 \tag{1}$$

The normalized sensor feature matrix is represented by $\mathbf{Z}$, and $\mathbf{p}_1$ refers to the first principal component vector.The resulting Health Index (HI) is then normalized to the range [0,1], ensuring a consistent depiction of degradation across various engine units and operating conditions.

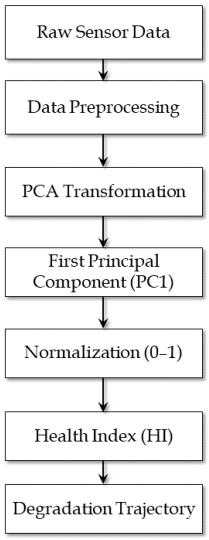


Fig. 3: Health Index Construction

3.7 Evaluation Metrics

The performance of the proposed Temporal Embedding-based Temporal Convolutional Neural Network (TE-TCNN) is evaluated using two metrics: Root Mean Square

Error (RMSE) and the NASA Scoring Function. RMSE is used to measure the difference between the predicted and actual RUL values, while the NASA score provides an additional measure of the prediction error.

RMSE represents the overall magnitude of the prediction error. It is calculated by taking the square root of the mean squared difference between the predicted and actual RUL values. A lower RMSE indicates that the predicted RUL values are closer to the actual values and therefore represents better prediction accuracy.

$$RMSE = \sqrt{\frac{1}{M}\sum_{j=1}^{M}(R_j - \tilde{R}_j)^2} \tag{2}$$

Where R_j represents the actual Remaining Useful Life (RUL), $tildeR_j$ is the predicted RUL, and M is the total number of test samples.A smaller RMSE value suggests greater prediction accuracy and closer alignment between the predicted and actual RUL values.

In addition to RMSE, the NASA Scoring Function is used to assess the practical importance of prediction errors. Unlike RMSE, the NASA score applies asymmetric penalties to early and late predictions, assigning a substantially higher penalty to late predictions since delayed maintenance decisions may result in unexpected equipment failures. Consequently, a lower NASA score indicates more reliable and practically effective prognostic performance.

$$Score = \sum_{j=1}^{M} \begin{cases} e^{-\frac{\Delta_j}{13}} - 1, & \Delta_j < 0 \\ e^{\frac{\Delta_j}{10}} - 1, & \Delta_j \geq 0 \end{cases} \tag{3}$$

where, Δ_j=$\tilde{R}_j$-R_j, represents the prediction deviation. Lower values of the NASA score correspond to improved model performance.

4 Results

4.1 Experimental Setup

The proposed Temporal Embedding-based Temporal Convolutional Neural Network (TE-TCNN) implemented with specific parameters Table 2 was implemented using the PyTorch deep learning framework and evaluated on the four benchmark subsets (FD001-FD004) of the NASA C-MAPSS dataset. These subsets represent different combinations of operating conditions and fault modes, making them suitable for evaluating the robustness and generalization capability of RUL prediction models. Following the standard practice in the literature, the Remaining Useful Life (RUL) values were truncated to a maximum of 125 cycles to avoid excessively large target values during the early stages of degradation.

Prior to model training, the sensor data underwent a series of preprocessing steps. Sensors exhibiting little or no variation were discarded, and the operating conditions were grouped using the K-means clustering algorithm. Cluster-wise standardization was then applied to minimize the influence of varying operating regimes and improve the consistency of the sensor measurements. Finally, a sliding window technique

Table 2: Hyperparameters of TE-TCNN

Hyperparameter	**Value**
Optimizer	Adam
Loss Function	Mean Squared Error (MSE)
Learning Rate	0.001 (FD001–FD003), 0.0001 (FD004)
Batch Size	128
Maximum RUL Threshold	125
Sliding Window Length	30 (FD001–FD003), 60 (FD004)
Training Epochs	20 (FD001), 25 (FD002), 20 (FD003), 45 (FD004)
Number of Convolutional Layers	2
Convolution Filters	64, 128
Kernel Size	5
Activation Function	ReLU
Batch Normalization	After each convolution layer
Dropout Probability	0.2
Gradient Clipping	1.0
Temporal Embedding Initialization	$\mathcal{N}(0, 0.01)$
Operating Condition Clusters	6 (K-means)
Feature Normalization	Cluster-wise Standardization

was employed to transform the multivariate time-series data into fixed-length input sequences, which were used as inputs to the proposed TE-TCNN model.

The TE-TCNN model was trained using the Adam optimization algorithm by minimizing the Mean Squared Error (MSE) loss function. The training hyperparameters were selected empirically based on preliminary experiments to achieve stable convergence and consistent predictive performance across all four datasets.The complete training configuration is summarized in Table 2.

4.2 RUL Prediction Performance

The comparative performance analysis of the proposed model was carried out on all four subsets of the C-MAPSS dataset using RMSE and NASA scoring metrics. Table 3 presents the performance comparison between the proposed approach and several existing deep learning-based RUL prediction models.

The obtained results indicate that the proposed model achieves competitive prediction accuracy across different operating conditions and fault modes. In particular, the model demonstrates stable performance on the more complex FD002 and FD004 datasets, which contain multiple operating conditions and fault combinations. Although certain CNN-based and hybrid approaches achieve lower RMSE values on specific subsets, the proposed method maintains consistent degradation tracking capability across all datasets.

It can also be observed that the proposed approach outperforms several conventional and recurrent learning-based methods such as Deep LSTM, GAN-based models, Heimes RNN, and Babu's CNN in terms of overall prediction accuracy. This indicates

Table 3: Comparative analysis of different RUL prediction models on the C-MAPSS datasets

Model	FD001		FD002		FD003		FD004	
	RMSE	Score	RMSE	Score	RMSE	Score	RMSE	Score
Conventional and Deep Learning Methods								
Deep LSTM [5]	16.14	338	24.49	4450	16.18	852	28.17	5550
Babu CNN [7]	18.45	1286.7	30.29	13570	19.82	1596.2	29.16	7886.4
Li CNN [8]	**12.61**	**273.7**	22.36	10412	12.64	284.1	23.31	12466
GAN Model [33]	18.54	1467	20.06	4085	19.28	1488	20.88	3872
TCMN [11]	23.57	1220	20.45	3100	21.17	1300	21.03	4000
MODBNE [2]	15.04	334.23	25.05	5585.34	12.51	421.91	28.66	6557.62
KDNet [17]	13.68	362.08	**14.47**	929.20	12.95	327.27	15.96	1303.19
Other Existing Baseline Methods								
Heimes RNN [4]	22.00	1050	31.20	6400	23.40	1350	32.10	7200
FTT-GRU [16]	30.76	2100	28.42	5800	24.63	1900	29.84	6100
PE-Net [15]	13.98	280.87	14.69	**881.73**	12.33	**272.85**	**15.40**	**1103.18**
TE-TCNN (Proposed Method)	16.636	608.552	17.096	1661.307	18.217	1508.267	19.150	2289.280

Lower RMSE and NASA score indicate better prediction accuracy. Reported baseline results are obtained from their respective studies.

that the adopted feature extraction and temporal learning strategy effectively capture degradation-related information from multivariate sensor signals.

Furthermore, the RMSE values achieved by the proposed model remain within an acceptable range for all operating conditions, demonstrating the robustness and generalization capability of the framework. However, some datasets show higher NASA scores because the scoring function uses an asymmetric penalty system, which applies much bigger penalties for late RUL predictions.Even with these challenges, the model still gives accurate estimates of degradation trends and keeps its prediction results stable.

A comparison shows that the proposed framework provides good predictive performance and works consistently in both simple and complex turbofan engine degradation scenarios.

4.3 Kernel Sensitivity Analysis

The effect of using different kernel sizes on model performance was studied to see how the size of the receptive field influences results.For this, the same kernel sizes of 3, 5, and 7 were used consistently in all convolutional layers, keeping the model structure the same but changing the size of the temporal receptive field.

The size of the convolutional kernel affects how well models can predict results on different C-MAPSS datasets which is observed in Table 4.When a kernel size of 3 is used, the TE-TCNN model performs better than both CNN and CNN-C in terms of prediction accuracy on FD001, FD002, and FD004, as shown by lower RMSE values.On FD003, TE-TCNN performs similarly to CNN, which means it can still learn

Table 4: Kernel Sensitivity Analysis (RMSE)

Dataset	Kernel	CNN	CNN-C	TE-TCNN
FD001	3	17.430	40.198	16.340
	5	17.489	19.622	16.838
	7	17.780	20.745	17.249
FD002	3	18.274	43.336	18.027
	5	17.059	17.869	16.853
	7	17.483	19.843	17.239
FD003	3	19.183	40.928	19.432
	5	18.798	19.645	19.111
	7	18.331	18.444	18.808
FD004	3	22.187	44.005	20.106
	5	21.218	20.099	18.999
	7	21.034	21.758	18.979

degradation patterns even when fault conditions change.The NASA scores also support this, as TE-TCNN shows lower prediction penalties compared to the baseline models.This suggests that the model gives more consistent RUL estimates by capturing how things degrade over time.In contrast, CNN-C has higher RMSE and NASA scores, showing that making the network more complex always does not improve prognostic performance.

When the kernel size is set to 5, TE-TCNN continues to perform well across all four C-MAPSS subsets. It achieves the lowest RMSE on FD001 and FD004, and performs similarly to CNN on FD002 and FD003.The NASA scores also remain favorable, with TE-TCNN achieving lower or comparable results compared to the baseline models.This demonstrates that the temporal embedding mechanism allows the model to effectively extract degradation features, regardless of the size of the receptive field. As a result, TE-TCNN offers stable and reliable RUL predictions for both early and advanced stages of degradation.Overall, a kernel size of 5 provides a balanced receptive field that captures both local and broader temporal dependencies effectively.

When the kernel size is increased to 7, the performance of TE-TCNN becomes more inconsistent. While it continues to maintain competitive RMSE values on FD001 and FD004, its performance slightly declines on FD003. Similarly, the NASA score increases for the more complex datasets, indicating that excessively large kernels may smooth important degradation information and reduce prediction accuracy. CNN exhibits relatively stable behaviour across datasets, whereas CNN-C continues to show unstable performance with considerably higher prediction errors.

Overall, TE-TCNN demonstrates competitive or improved performance compared with the baseline CNN across the evaluated kernel configurations and substantially outperforms CNN-C in several cases. Among the evaluated kernel sizes, kernel size 5 provides the best overall trade-off between prediction accuracy, robustness, and temporal feature extraction, and is therefore selected as the optimal configuration for the proposed model.

Table 5: Kernel Sensitivity Analysis using NASA Score

Dataset	Kernel	CNN	CNN-C	TE-TCNN
FD001	3	818.387	13897.319	595.663
	5	846.212	2381.716	759.030
	7	903.574	1624.715	821.424
FD002	3	1615.358	56960.550	2442.144
	5	1414.448	1994.402	1544.026
	7	1377.729	2778.526	1816.582
FD003	3	1815.251	27547.734	1988.624
	5	1278.840	1738.385	1852.798
	7	1319.442	845.844	1419.075
FD004	3	3867.875	94067.960	3436.405
	5	2877.062	3516.568	2420.806
	7	2625.264	11852.009	2081.372

The analysis of NASA scores further shows how the size of the kernel affects the stability and speed of predictions results imprinted in Table 5.Both CNN and CNN-C show noticeable changes in NASA scores depending on the kernel settings, especially in the more complex datasets FD002 and FD004.When larger kernels are used, CNN-C becomes less stable, as shown by the much higher NASA scores.These higher scores suggest that the model has difficulty accurately predicting the remaining useful life, particularly in the later stages of degradation, which causes the predictions to be delayed.In contrast, the proposed TE-TCNN model keeps the NASA scores lower and more consistent across most datasets, showing better reliability and stability.This suggests that the learnable temporal embedding in TE-TCNN helps it better understand how the degradation progresses over time, reducing the need for larger convolutional kernels to extract temporal information.

To better explain how the kernel size affects model performance, graphical comparisons of RMSE and NASA scores are shown for each kernel setup across all four C-MAPSS datasets.These visuals also show how consistently TE-TCNN performs under different receptive field configurations and demonstrate how the prediction behavior varies among the models being compared. In accordance with Fig. 4, CNN-C shows much higher RMSE values on all datasets, showing that it does not generalize well even though it has a deeper network structure.These findings show that using a learnable temporal embedding helps in better feature representation and allows for more accurate modeling of degradation compared to traditional convolutional architectures.

Fig 5 denotes NASA score comparison for kernel size 3. TE-TCNN consistently achieves the lowest or near-lowest NASA scores across the datasets, indicating fewer severe prediction penalties. CNN also provides competitive performance, whereas CNN-C exhibits extremely high NASA scores, particularly for FD002 and FD004, reflecting unstable and less reliable predictions. The results suggest that temporal

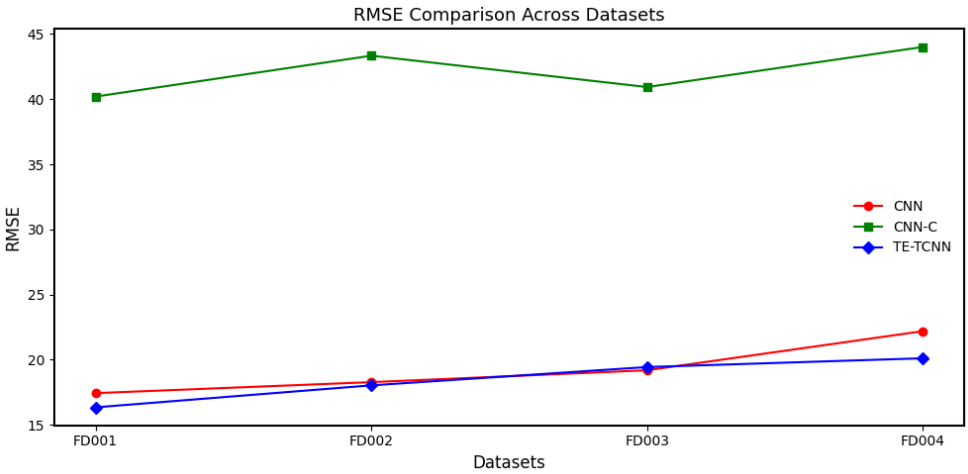


Fig. 4: RMSE for kernel size 3

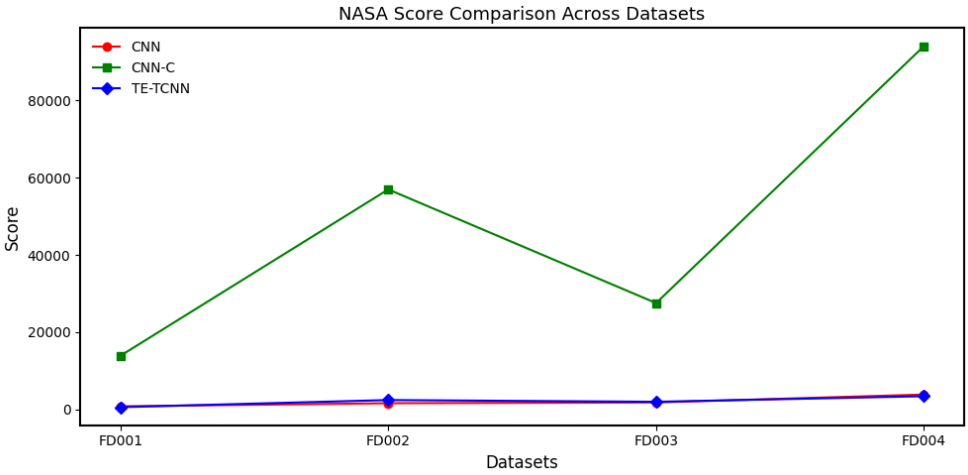


Fig. 5: Score of Kernel 3

embedding enables TE-TCNN to capture degradation progression more effectively while maintaining stable prediction behaviour.

Illustrated in Fig 6 the RMSE comparison for kernel size 5. TE-TCNN achieves the lowest RMSE on FD001 and FD004 and remains highly competitive on FD002 and FD003. CNN exhibits similar performance on FD002 and FD003, whereas CNN-C records comparatively larger prediction errors on most datasets. These results indicate that kernel size 5 provides an effective receptive field for extracting degradation-related temporal features while maintaining stable prediction accuracy.

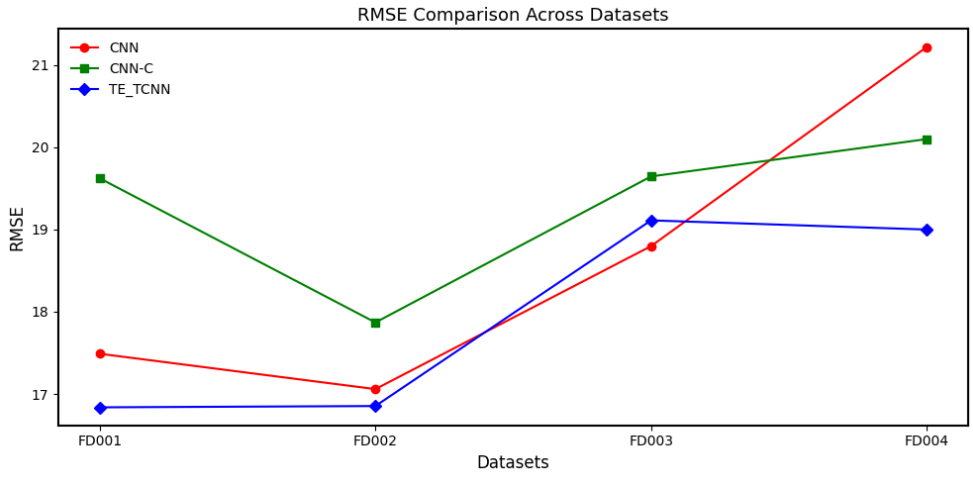


Fig. 6: RMSE of Kernel 5

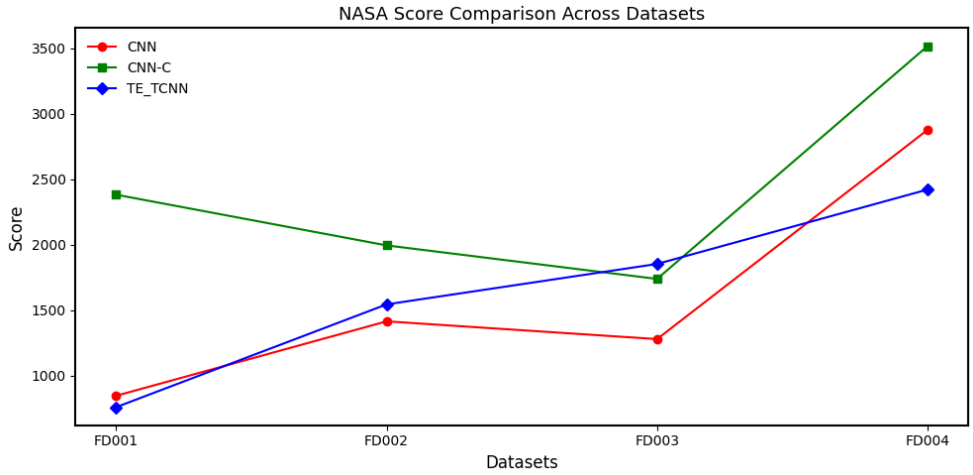


Fig. 7: Score of Kernel 5

Fig. 7 compares the NASA scores for kernel size 5. TE-TCNN maintains consistently low prediction penalties across all datasets and performs better than CNN, particularly on FD004. CNN-C continues to produce higher NASA scores, indicating reduced prediction reliability under complex operating conditions. The results further confirm that the proposed temporal embedding improves temporal feature learning and prediction stability.

Demonstrated in Fig. 8 presents the RMSE comparison for kernel size 7. Although TE-TCNN remains competitive on FD001 and FD004, its performance slightly

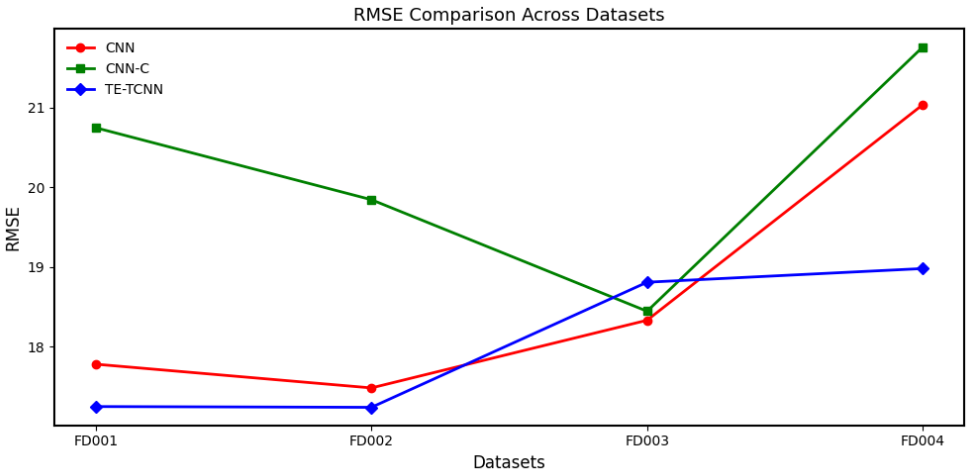


Fig. 8: RMSE of Kernel 7

degrades on FD003 compared with smaller kernel sizes. CNN performs comparably on FD001 and FD002, while CNN-C again records the highest RMSE values on most datasets. These observations indicate that increasing the kernel size beyond an optimal value does not necessarily improve feature extraction and may reduce prediction accuracy for certain degradation patterns.

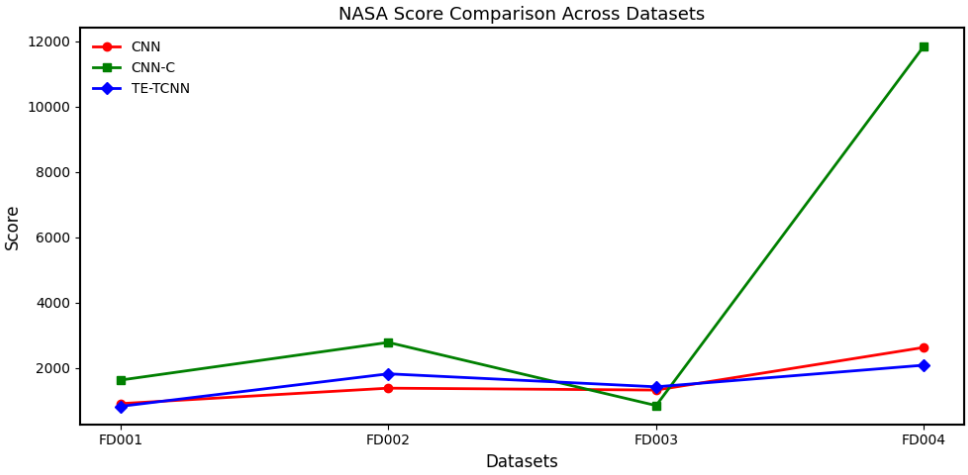


Fig. 9: Score of Kernel 7

Fig 9 establishes the NASA scores compare for different models using a kernel size of 7 on all the C-MAPSS datasets.TE-TCNN keeps showing stable prediction results,

but there is a slight increase in NASA scores when the operating conditions are more complicated.The CNN model behaves in a similar way, with some moderate drops in prediction accuracy, while CNN-C has a big drop in performance, especially for FD004, where the NASA score goes up a lot.This suggests that making the convolutional kernel bigger does not always help in capturing better temporal features, as larger kernels might make it harder for the model to pick up on small signs of wear and tear. The overall results show that the size of the convolutional kernel is a key factor in how accurate are RUL predictions. Among the tested setups, a kernel size of 5 offers the best balance between capturing short-term degradation changes and learning long-term temporal patterns.Although a kernel size of 3 gives results that are nearly as good, its smaller receptive field may not allow for the extraction of broader degradation trends.In contrast, a kernel size of 7 leads to more inconsistent prediction performance, likely because it smooths out key degradation-related features too much.Across all kernel sizes, TE-TCNN consistently produces more stable and dependable predictions compared to CNN, and it clearly outperforms CNN-C.These results confirm that adding a learnable temporal embedding mechanism improves the ability of temporal convolutional networks to model complex degradation behaviors, without needing to rely only on increasing the size of the convolutional filters.

4.4 Model Comparison

Table 6: Performance Comparison Using Selected Architecture-Specific Kernel Configurations

Dataset	Model	Kernel Size	RMSE	NASA Score
FD001	CNN	(3,3)	17.430	818.387
	CNN-C	(7,5,5)	19.463	1034.025
	TE-TCNN	(5,5)	16.686	723.195
FD002	CNN	(3,3)	18.355	1540.240
	CNN-C	(7,5,5)	18.117	3645.925
	TE-TCNN	(5,5)	16.686	1501.296
FD003	CNN	(3,3)	19.133	1831.799
	CNN-C	(7,5,5)	19.700	1232.325
	TE-TCNN	(5,5)	19.071	1861.195
FD004	CNN	(3,3)	21.749	3412.698
	CNN-C	(7,5,5)	20.795	6813.094
	TE-TCNN	(5,5)	19.760	3433.592

Table 6 compares the performance of CNN, CNN-C, and the proposed TE-TCNN using their respective kernel configurations across the four C-MAPSS datasets. These configurations were selected based on the preceding kernel sensitivity analysis and represent the final configurations used for architecture-level comparison. Specifically, CNN employs a kernel configuration of (3,3), CNN-C uses (7,5,5), and TE-TCNN adopts (5,5). The results demonstrate that the proposed TE-TCNN provides competitive prediction performance across different operating conditions. In particular,

TE-TCNN achieves the lowest RMSE on FD001, FD002, and FD004, while its performance on FD003 remains comparable to the baseline models. These observations indicate that integrating learnable temporal embedding with a moderate kernel configuration enables effective temporal feature extraction without substantially increasing model complexity.

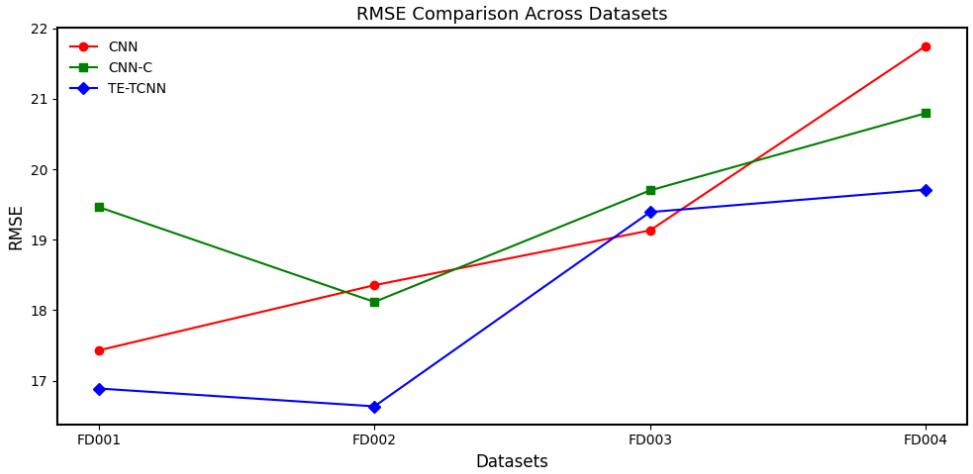


Fig. 10: RMSE comparison across all machine learning models

Fig 10 presents the RMSE comparison of the three models across the four datasets. TE-TCNN records the lowest RMSE for FD001 and FD002, demonstrating its ability to accurately model degradation behaviour in relatively simpler operating conditions. On FD003, the RMSE values of all three models are close, with CNN exhibiting a marginally lower error than TE-TCNN. For the more challenging FD004 dataset, TE-TCNN again achieves the lowest RMSE, outperforming both CNN and CNN-C. In contrast, CNN-C exhibits greater variation in prediction error across datasets, indicating that increasing network depth and kernel size alone does not necessarily improve prediction accuracy. Overall, the results suggest that the proposed TE-TCNN provides a favourable balance between prediction accuracy and model complexity.

The standard CNN employing the (3,3) kernel configuration demonstrates relatively stable performance across all datasets. However, its smaller receptive field limits its ability to capture longer-range temporal degradation characteristics, resulting in slightly higher prediction errors than TE-TCNN in most cases. Although the CNN-C architecture incorporates deeper convolutional layers with larger kernels, it does not consistently improve performance. The increased model complexity appears to introduce greater variability, particularly on FD001 and FD004, where CNN-C records noticeably higher RMSE values.

An interesting observation can be made for FD003, where the RMSE values of all three models are relatively close. This suggests that the degradation characteristics

of this dataset can be effectively captured by different convolutional architectures, reducing the performance gap between the proposed model and the baseline methods.

Overall, the mixed-kernel comparison demonstrates that the proposed TE-TCNN achieves a more consistent balance between prediction accuracy and robustness than the baseline CNN architectures. The combination of learnable temporal embedding with the selected (5,5) kernel configuration enables effective extraction of temporal degradation information while maintaining a lightweight network architecture.

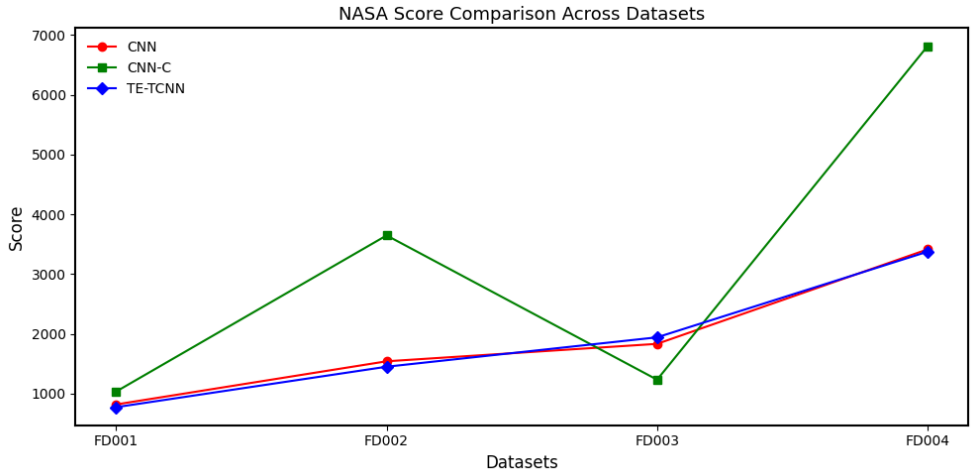


Fig. 11: Score comparison across all machine learning models

Fig 11 compares the NASA scores obtained by the three models under the mixed-kernel configuration. Similar to the RMSE analysis, TE-TCNN achieves the lowest NASA scores on FD001 and FD002, indicating fewer severe prediction penalties under relatively simple operating conditions. On FD003, the TE-TCNN score is slightly higher than that of CNN-C but remains comparable to the standard CNN. For FD004, TE-TCNN and CNN produce similar NASA scores, whereas CNN-C exhibits a substantial increase in prediction penalty, indicating poor generalization under complex operating conditions.

The CNN model demonstrates a gradual increase in NASA score as dataset complexity increases, reflecting relatively stable prediction behaviour across the four datasets. In contrast, CNN-C exhibits considerable fluctuations, particularly on FD002 and FD004, where the prediction penalties increase substantially despite the deeper architecture. These results indicate that simply increasing network depth and kernel size does not necessarily improve prediction reliability.

Overall, the proposed TE-TCNN maintains competitive prediction performance across all datasets while providing a better balance between accuracy and robustness than the deeper CNN-C model. The learnable temporal embedding enables more effective modelling of degradation progression, contributing to improved prediction stability without introducing excessive architectural complexity.

4.5 RUL Prediction Analysis

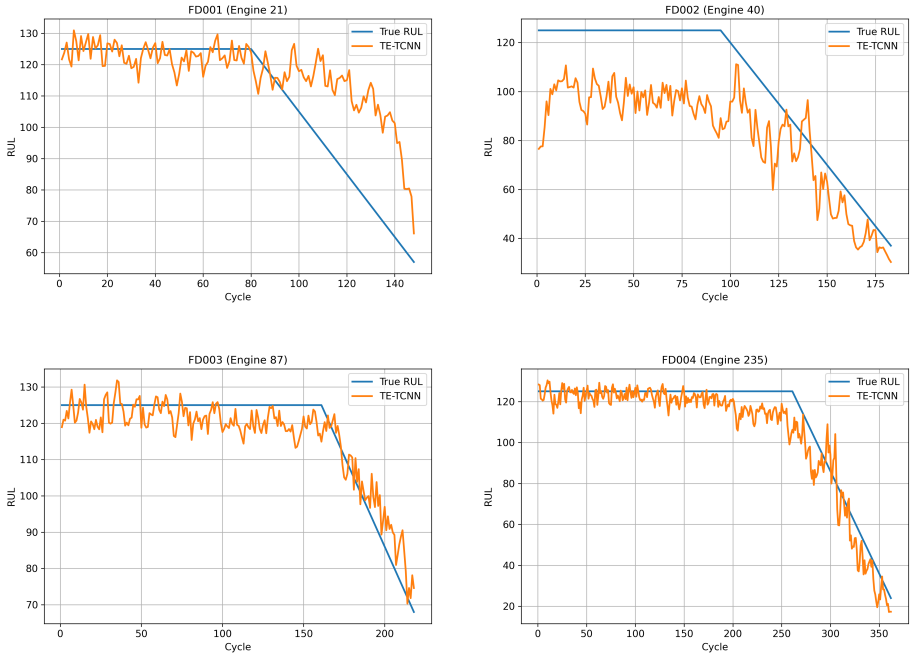


Fig. 12: True RUL versus predicted RUL for representative engines from the FD001–FD004 datasets

The prediction results for selected engines from the four C-MAPSS subsets (FD001–FD004) are presented in Fig. 12. The predicted Remaining Useful Life (RUL) is compared with the corresponding true RUL values to examine how the proposed TE-TCNN model follows the degradation progression. The results are also considered in relation to the PE-Net model reported by Jin et al. [15], which investigates RUL prediction under varying operating conditions.

For FD001, the predicted RUL follows the overall decrease in the true RUL as the engine moves from the earlier operating stage toward failure. The difference between the two curves becomes more noticeable at some cycles, but the main degradation trend is maintained. This indicates that the model is able to extract useful temporal information from the sequence of sensor observations.

For FD002, the prediction curves show greater variation than those observed for FD001. This behaviour is expected from the presence of multiple operating conditions in the dataset. Despite these variations, the predicted RUL generally follows the direction of the true RUL curve, indicating that the temporal features learned by TE-TCNN remain useful when the operating conditions change.

For FD003, the predicted and true RUL curves show a similar overall progression over the selected engine life. Some local differences can be observed between the two

curves, particularly in portions of the degradation period. However, the prediction maintains the general downward trend of the true RUL, including the later degradation stage.

FD004 presents the most challenging prediction case because it combines multiple operating conditions with multiple fault modes. The corresponding prediction curves show greater variation than those of the other subsets. Even under these conditions, the predicted RUL follows the main trend of the true values for the selected engines. The result suggests that the proposed model can retain useful temporal information when the degradation behaviour becomes more variable.

The prediction plots show that TE-TCNN follows the general RUL progression across all four datasets. The differences between the predicted and true curves vary between engine units and operating conditions. The use of a learnable temporal embedding provides the model with an additional trainable representation of the temporal position of observations before convolutional feature extraction.

4.6 Health Index and Degradation Analysis

The Health Index (HI) is used to represent the change in engine condition over its operating life. In this study, the HI is normalized between 0 and 1, where higher values represent a healthier condition and lower values indicate greater degradation. The HI trajectories are examined across engine cycles to observe how the condition changes as the engines approach failure.

In general, the HI trajectories show a gradual change during the earlier part of the engine life, followed by a more noticeable decline in the later cycles. This behaviour provides a visual indication of the progression from relatively healthy operation to the degraded state. Small variations between individual engine trajectories can also be observed. These variations may be related to differences in sensor measurements and operating conditions across the engine units.

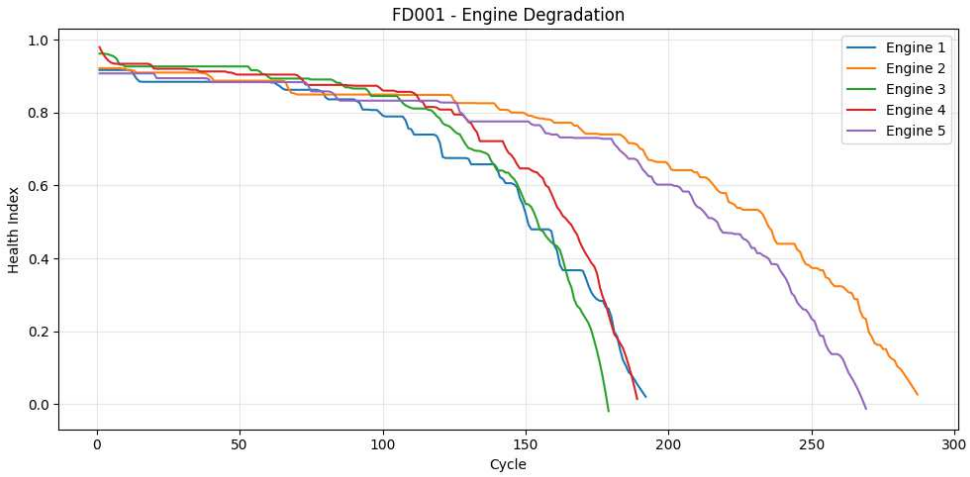


Fig. 13: Health degradation trajectories of FD001

Fig. 13 shows the HI trajectories for selected engine units from the FD001 dataset. The trajectories remain relatively stable during the initial cycles and gradually decrease as the engines progress through their operating life. A more pronounced decline can be observed near the later cycles. Since FD001 contains a single operating condition and a single fault mode, the trajectories show comparatively less variation between the selected engines.

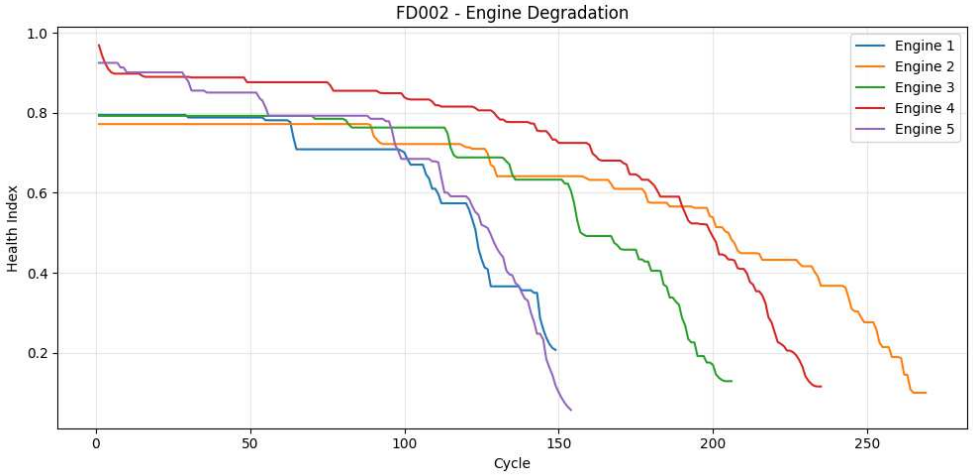


Fig. 14: Health degradation trajectories of FD002

Fig. 14 presents the HI trajectories for FD002. Compared with FD001, greater variation can be observed between the individual trajectories. This is consistent with the presence of multiple operating conditions in this subset, which can lead to differences in the sensor measurements. Despite this variation, the trajectories generally show a downward trend as the engines progress toward failure.

Fig.fig:health-degradation-fd003 shows the HI trajectories for FD003. The trajectories exhibit more variation than those observed for FD001. This can be associated with the presence of two fault modes in the dataset. Although the degradation paths are not identical for all engine units, the HI generally decreases with increasing engine cycles, indicating a gradual change in the estimated health condition.

Fig.fig:health-degradation-fd004 presents the HI trajectories for FD004, which combines multiple operating conditions and fault modes. The trajectories show greater variation compared with the other subsets. Nevertheless, a general decline in HI can be observed as the engines approach the later stages of their operating life. The variation in the trajectories also reflects the increased complexity of the operating and fault conditions present in FD004.

4.7 Analysis on FD004 Dataset

For FD004, making predictions is harder because the data covers many operating conditions and different kinds of faults.This leads to a more complex and varied way the engine's performance declines.Unlike simpler datasets, the way the engine degrades

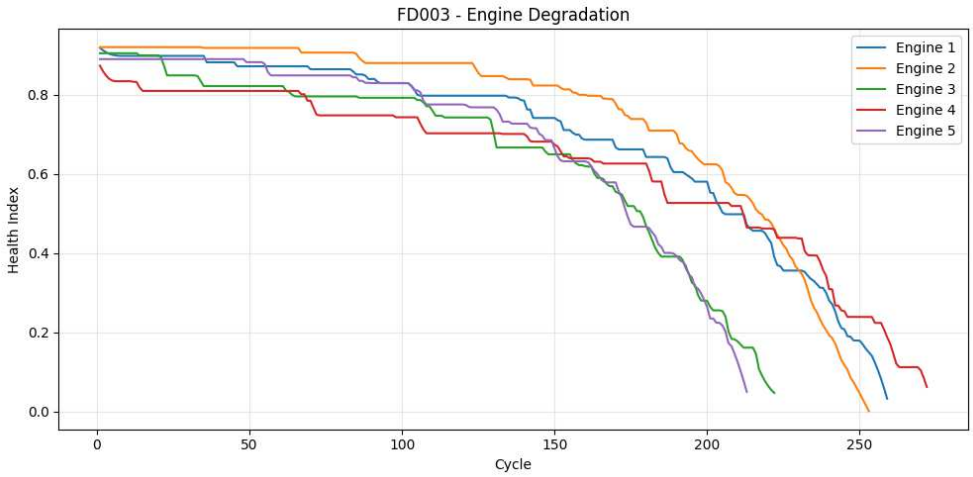


Fig. 15: Health degradation trajectories of FD003

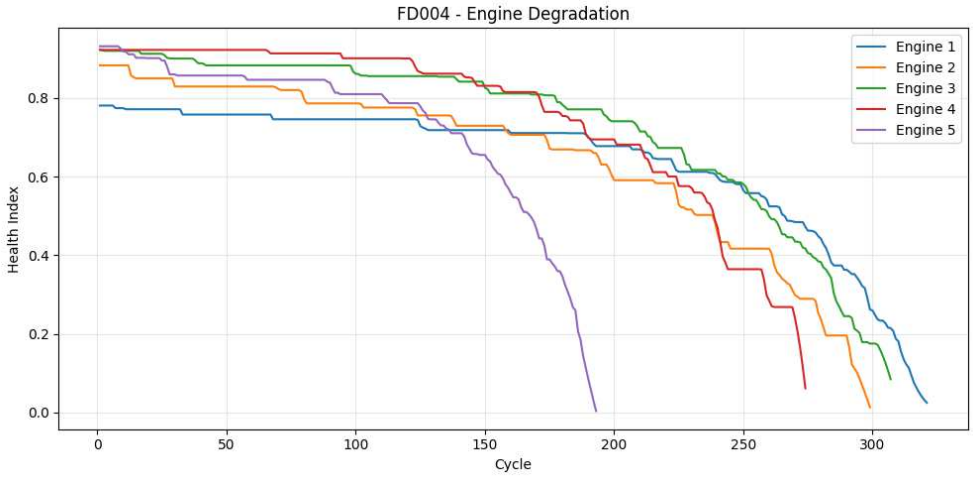


Fig. 16: Health degradation trajectories of FD004

in FD004 isn't the same every time, which makes it harder for prediction models to track how the engine's condition changes over time.

In such tough situations, the proposed TE-TCNN model shows steady and reliable results. It can detect both slow, ongoing changes in the engine's performance and sudden spikes in sensor readings.This helps the model better represent how degradation actually happens.This ability allows TE-TCNN to closely follow the real remaining useful life (RUL) of the engine, especially in the later stages when degradation speeds up and failure-related changes become more obvious.

Compared to traditional CNN and CNN-C models, TE-TCNN offers smoother and more stable RUL predictions with fewer sharp changes. This suggests that including a learnable temporal embedding helps the model better capture time-based degradation patterns from sequences of sensor data, rather than just focusing on short-term features.

5 Future Works

The TE-TCNN framework delivers positive outcomes when used to predict the Remaining Useful Life (RUL), but there's still room for growth. Incorporating TE-TCNN with other models, like LSTM, GRU, or even attention-based transformer networks, may help to identify complex degradation patterns.The Health Index (HI) model could be enhanced by the integration of Physics of Failure (PoF) and other similar domain-related information.Implementation of Bayesian deep learning in the model may help to establish uncertainties in predicting the RUL of a device and consequently help in decision-making for device maintenance.

Advancements could include the improvement of TE-TCNN in real-world applications, in scenarios that include high levels of uncertainty, such as incomplete information, missing data, and even sensor noise.Domain adaptation and transfer learning may help the model to easy adjust to different cases of degradation, operating conditions, and engines of different types.Even though the NASA C-MAPSS dataset is very popular for the evaluation of RUL prediction frameworks, comparing TE-TCNN to other frameworks on real-world industrial data may help provide a more accurate real-world assessment of the model.Improvements may focus on future work that aims to reduce the computational burden and increase adaptability while also improving systems, especially real-time systems.Other related data sources, such as operational logs and environmental data, could help add to the model and improve the confidence level of TE-TCNN in building prognostic systems.

6 Conclusion

In this study, a novel Temporal Embedding-based Temporal Convolutional Network (TE-TCNN) architecture is introduced to predict the Remaining Useful Life (RUL) of aircraft engines, based on the NASA C-MAPSS benchmark dataset. The framework integrates a learnable temporal embedding technique and a lightweight temporal convolution structure, making it capable of efficiently learning the diverse and changing patterns of degradation in multivariate sensor data. Also, a procedure based on PCA is introduced to construct a Health Index (HI) by imposing smoothing and monotonicity constraints to ensure a stable and interpretable description of the degradation of an engine.The kernel sensitivity analysis is presented here to understand the receptive field of convolutions and its impact on the predictive performance of the model.

Compared to the existing methods, the obtained results for all four C-MAPSS subsets (FD001-FD004) demonstrated that TE-TCNN is capable of maintaining a considerable compactness of convolution while achieving satisfactory RMSE and NASA Score.The model demonstrated robustness in a wide range of degradation conditions,

and in particular for FD004, where identify RUL estimation becomes challenging due to complex and rapidly changing degradation patterns.

TE-TCNN exemplifies a good solution for achieving accurate predictions while focusing on simplicity and efficiency in architecture and prognostics and health management (PHM) in general, as well as in prognostics.

In the future, we aim to address complete and thorough uncertainty-aware RUL prediction and extend the approach to different industrial datasets.

Declarations

Funding
No funding was received for conducting this study or for the preparation of this manuscript.

Competing Interests
The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data Availability
The C-MAPSS turbofan engine degradation dataset used in this study is publicly available through the NASA Prognostics Center of Excellence repository. The experiments were conducted using the FD001–FD004 subsets of the C-MAPSS dataset, originally described by Saxena et al. [1].

Code Availability
The code used to implement the proposed TE-TCNN model and conduct the experiments is available from the corresponding author upon reasonable request.

Author Contributions
Aditi Gawde contributed to the methodology, implementation, experimentation, data analysis, and manuscript preparation. Avinash More contributed to the research direction, supervision, interpretation of results, and critical revision of the manuscript. Both authors reviewed and approved the final manuscript.

Ethics Approval
No participants were involved in this study. Hence, ethical approval, informed consent, and adherence to institutional or licensing regulations are not applicable.

Consent to Participate
Consent to participate was not applicable to this study, as the research did not involve human participants, human subjects, or the collection of personal information.

Consent for Publication
Consent for publication was not applicable to this study, as no individual-level personal data, identifiable information, or images of human participants were used or

reported.

References

[1] Saxena, A., Goebel, K.: Turbofan engine degradation simulation data set. NASA Ames Prognostics Data Repository (2008)

[2] Lei, Y., Li, N., Lin, J.: A new method based on stochastic process models for machine remaining useful life prediction. IEEE Transactions on Instrumentation and Measurement **65**(12), 2671–2684 (2016)

[3] Zhang, C., Lim, P., Qin, A., Tan, K.: Multiobjective deep belief networks ensemble for remaining useful life estimation in prognostics. IEEE Transactions on Neural Networks and Learning Systems **28**(10), 2306–2318 (2017)

[4] Heimes, F.O.: Recurrent neural networks for remaining useful life estimation. In: International Conference on Prognostics and Health Management (2008)

[5] Zheng, S., Ristovski, K., Farahat, A., Gupta, C.: Long short-term memory network for remaining useful life estimation. In: IEEE International Conference on Prognostics and Health Management, pp. 88–95 (2017)

[6] Zhao, R., Wang, J., Yan, R., Mao, K.: Machine health monitoring using local feature-based gated recurrent unit networks. IEEE Transactions on Industrial Electronics **65**(2), 1539–1548 (2018)

[7] Babu, T., Zhao, P., Li, X.: Deep convolutional neural network based regression approach for estimation of remaining useful life. In: International Conference on Database Systems for Advanced Applications, pp. 214–228 (2016)

[8] Li, X., Ding, Q., Sun, J.Q.: Remaining useful life estimation in prognostics using deep convolution neural networks. Reliability Engineering & System Safety **172**, 1–11 (2018)

[9] Li, Y., Wang, X., Chen, L.: Multi-scale deep convolutional neural network for remaining useful life prediction. IEEE Transactions on Industrial Electronics **67**(12), 10303–10312 (2020)

[10] Li, S., Wang, K., Liu, J.: Ensemble learning based convolutional neural networks for remaining useful life prediction. Mechanical Systems and Signal Processing **133** (2019)

[11] Ren, H., Liu, M., Yang, Z., Yin, Y.: Temporal convolutional memory networks for remaining useful life estimation of industrial machinery. IEEE Transactions on Industrial Informatics **16**(6), 4097–4107 (2020)

[12] Vaswani, A., *et al.*: Attention is all you need. In: Advances in Neural Information Processing Systems, pp. 5998–6008 (2017)

[13] Zhou, H., *et al.*: Informer: Beyond efficient transformer for long sequence time-series forecasting. In: AAAI Conference on Artificial Intelligence, vol. 35, pp. 11106–11115 (2021)

[14] Liu, X., Zhang, Y., Zhao, H.: Ftt-gru: A hybrid fast temporal transformer-gru framework for remaining useful life prediction. IEEE Access **10**, 45812–45824 (2022)

[15] Jin, Z., He, Y., Wei, X.: Pe-net: A position encoding network for remaining useful life prediction. Reliability Engineering & System Safety **214** (2021)

[16] Woo, S., Park, J., Lee, J., Kweon, I.S.: Cbam: Convolutional block attention module. In: European Conference on Computer Vision (ECCV), pp. 3–19 (2018)

[17] Zhang, Y., Wang, Q., Zhang, L.: Kdnet: Knowledge distillation network for remaining useful life prediction. IEEE Access **9**, 21862–21875 (2021)

[18] Raissi, M., Perdikaris, P., Karniadakis, G.E.: Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational Physics **378**, 686–707 (2019)

[19] Jiang, Z., Wei, Y., Li, X.: Remaining useful life prediction across different operational conditions based on domain adaptation feature transfer. Scientific Reports (2026)

[20] Wang, Y., Su, C., Wang, P., Zhen, J., Wang, D.: A hybrid large-kernel cnn and markov feature framework for remaining useful life prediction. Machines **14**(1), 57 (2026)

[21] Zhang, T., Jiang, L., Huang, R., Zhang, X.: A multi-scale cross-channel attention network for remaining useful life prediction with variable sensors. IEEE Transactions on Instrumentation and Measurement (2025)

[22] Wen, Z., Fang, Y., Wei, P., Liu, F., Chen, Z., Wu, M.: Temporal and heterogeneous graph neural network for remaining useful life prediction. IEEE Transactions on Neural Networks and Learning Systems (2025)

[23] Kim, E., Park, S., Lee, H., Ko, S., Hwang, E.: Enhanced remaining useful life prediction for turbofan engines using spatio-temporal koopman dual-branch transformer. IEEE Transactions on Instrumentation and Measurement (2025)

[24] Chen, X.: A novel transformer-based dl model enhanced by position-sensitive attention and gated hierarchical lstm for aero-engine rul prediction. Scientific

Reports **14**(1), 10061 (2024)

[25] Sun, S., Wang, J., Xiao, Y., Peng, J., Zhou, X.: Few-shot rul prediction for engines based on cnn-gru model. Scientific Reports **14**(1), 16041 (2024)

[26] Liu, C.-L., Xiao, B., Hsu, S.-S.: Self-supervised learning for remaining useful life prediction using simple triplet networks. Advanced Engineering Informatics **64**, 103038 (2025)

[27] Niu, T., Xu, Z., Luo, H., Zhou, Z.: Hybrid gaussian process regression with temporal feature extraction for partially interpretable remaining useful life interval prediction in aeroengine prognostics. Scientific Reports **15**(1), 11057 (2025)

[28] Ness, S.: Enhancing remaining useful life prediction against adversarial attacks: An active learning approach. IEEE Access (2025)

[29] Wu, F., Wu, Q., Tan, Y., Xu, X.: Remaining useful life prediction based on deep learning: A survey. Sensors **24**(11), 3454 (2024)

[30] Li, W., Chen, J., Chen, S., Li, P., Zhang, B., Wang, M., Yang, M., Wang, J., Zhou, D., Yun, J.: A comprehensive review of artificial intelligence-based algorithms for predicting the remaining useful life of equipment. Sensors **25**(14), 4481 (2025)

[31] Wang, Y., Wu, M., Li, X., Xie, L., Chen, Z.: A survey on graph neural networks for remaining useful life prediction: Methodologies, evaluation and future trends. Mechanical Systems and Signal Processing **229**, 112449 (2025)

[32] Chen, Y., Liu, C.: Remaining useful life prediction: A study on multidimensional industrial signal processing and efficient transfer learning based on large language models. arXiv preprint arXiv:2410.03134 (2024)

[33] Zhang, Y., Yang, R., et al.: Generative adversarial network-based data augmentation for remaining useful life prediction. Mechanical Systems and Signal Processing **139** (2020)