Indonesian Journal of Electrical Engineering and Computer Science
Vol. 41, No. 1, January 2026, pp. 283∼299
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v41.i1.pp283-299

Remaining useful life estimation of turbofan engine: a sliding time window approach using deep learning

Alawi Alqushaibi[1], Mohd Hilmi Hasan[1,2], Said Jadid Abdulkadir[1,2], Shakirah Mohd Taib[1,2], Safwan Mahmood Al-Selwi[1], Mohammed Gamal Ragab[1], Ebrahim Hamid Sumiea[1]
[1]Department of Computer and Information Sciences, Universiti Teknologi PETRONAS, Seri Iskandar, Malaysia
[2]Centre for Research in Data Science (CERDAS), Universiti Teknologi PETRONAS, Seri Iskandar, Malaysia

Article Info

Article history:

Received Dec 1, 2023
Revised Nov 9, 2025
Accepted Dec 13, 2025

Keywords:

CMAPSS
Convolutional neural network
Deep features
Prognostics
Recurrent neural networks
RUL prediction
Turbofan engine

ABSTRACT

System degradation is a common and unavoidable process that frequently occurs in aerospace sector. Thus, prognostics is employed to avoid unforeseen breakdowns in intricate industrial systems. In prognostics, the system health status, and its remaining useful life (RUL) are evaluated using numerous sensors. Numerous researchers have utilized deep-learning techniques to estimate RUL based on sensor data. Most of the studies proposed solving this problem with a single deep neural network (DNN) model. This paper developed a novel turbofan engine RUL predictor based on several DNN models. The method includes a time window technique for sample preparation, enhancing DNN's ability to extract features and learn the pattern of turbofan engine degradation. Furthermore, the effectiveness of the proposed approach was confirmed using well-known model evaluation metrics. The experimental results demonstrated that among four different DNNs, the long short-term memory (LSTM)-based predictor achieved the better scores on an independent testing dataset with a root-mean-square error of 15.30, mean absolute error score of 2.03, and R-squared score of 0.4354, which outperformed the previously reported results of turbofan RUL estimation methods.

Corresponding Author:

Alawi Alqushaaibi
Department of Computer and Information Sciences, Universiti Teknologi PETRONAS
Seri Iskandar, 32610 Perak, Malaysia
Email: alawi_18000555@utp.edu.my

1. INTRODUCTION

Prognostics and Health Management (PHM) stands as a burgeoning discipline with the primary objective of predicting the prospective health condition of a given system, pinpointing latent faults, and facilitating punctual maintenance interventions to improve the reliability as well as the operational availability of the system [1]. As contemporary engineering systems, including aerospace, automotive, and manufacturing domains, continue to grow in complexity, there arises an escalating demand for advanced PHM methodologies adept at managing substantial datasets and furnishing precise prognostications [2], [3]. The concept of PHM has evolved over the past few decades, driven by the need to improve system reliability, safety, and efficiency [4]. Initially, PHM was mainly used in the aerospace industry to monitor the health of aircraft engines and to predict their remaining useful life [5]. With advancements in sensor technology, data analytics, and machine learning algorithms, the scope of PHM has expanded to other domains [6]. Today, PHM is applied in a wide range of applications, including wind turbines [7], medical devices, and infrastructure systems [8].

Despite the numerous advantages inherent in PHM, it is imperative to acknowledge the existence of several challenges that warrant attention. A predominant challenge resides in the absence of standardized practices within the discipline, a factor that introduces complexity in the comparative evaluation of diverse PHM methodologies [9]. An additional obstacle pertains to the requisite acquisition of extensive sets of superior data for the purpose of training predictive models. This process often incurs significant costs and consumes substantial time, thereby presenting a formidable challenge [10]. Additionally, PHM requires interdisciplinary expertise, which may not always be readily available [11]. Deep learning (DL), a subset within the realm of machine learning, has demonstrated remarkable promise in the domain of PHM owing to its capacity to comprehend intricate correlations between input characteristics and predictive outcomes. Over recent years, methodologies stemming from DL, including: convolutional neural networks (CNNs), recurrent neural networks (RNNs), and autoencoders, have found extensive application and utilization within PHM research endeavors [12].

Within the realm of fault diagnosis, the utilization of deep learning methodologies has been instrumental in discerning and categorizing faults by analyzing sensor data [13]. CNNs have been shown to be effective in feature extraction from sensor signals, while RNNs have been used to capture temporal dependencies between sensor measurements [14]. Autoencoders have also been used for fault detection by learning the normal operating conditions of a system and detecting deviations from these conditions [15]. In remaining useful life (RUL) prediction, DL models have been used to predict the RUL of a system based on its current and past health status [16]. CNNs and RNNs have been used to model the temporal evolution of system health, while autoencoders have been used to learn the underlying feature representations of sensor data [17]. Another important application of deep learning in PHM is anomaly detection [18]. DL models have been used to recognize abnormal behaviour in sensor data, which can indicate potential faults or anomalies [19]. CNNs and autoencoders have been shown to be effective in detecting anomalies in sensor data [10]. However, this study focuses on RUL predication which is in the third level of PHM [20].

This article endeavors to furnish a comparative analysis concerning prevalent deep learning architectures employed in prognostics for predicting RUL. Our emphasis will be on examining CNNs, RNNs, LSTM networks, and gated recurrent unit (GRU) models. The performance of these architectures will be evaluated based on prediction accuracy, computational complexity, and generalization ability to unseen data. Our aim is to provide practitioners and researchers with an inclusive overview of these architectures and their relative weaknesses and strengths for RUL prediction in prognostics. The validation of this methodology's effectiveness was conducted using the commercial modular aero-propulsion system simulation (C-MAPSS) turbofan aero-engine benchmark datasets supplied by NASA.

The subsequent sections of this manuscript are structured as follows: section 2 furnishes a comprehensive overview delineating the background and pertinent literature that form the foundation of this study. Section 3 delineates the suggested approach for conducting research, while section 4 examines and scrutinizes the obtained results from empirical experiments. Section 5 goes over the findings and analysis. Finally, Section 6 encapsulates the conclusions drawn from this study and delineates prospective avenues for future research.

2. RELATED WORK

In the aerospace sector, ensuring safety and reliability stands as a paramount consideration governing operational efficiency. Across various industries, rotating machinery assumes a pivotal role, yet remains susceptible to failure due to demanding operational environments and prolonged usage hours [21]. Failures within these systems can lead to operational disruptions and substantial financial ramifications. Exploring the monitored relationship between device data and its associated RUL has garnered significant attention in data-driven prognostics. Numerous machine learning algorithms, particularly NN methods, have been devised to unveil the correlation between the collected feature data and the anticipated RUL. The benefit of employing NNs for PHM lies in their ability to model intricate, highly nonlinear, multidimensional structures without a prior understanding of the system's physical behavior. Diverse forms of device data, like raw sensor readings, can serve as direct inputs for these models. However, establishing natural confidence limits for deep neural network (DNN) methodologies applied to prognostic issues demonstrate encouraging outcomes RUL prognostication remains a challenge [22], DNN-based approaches to prognostic problems show promising results [20].

Fentaye *et al.* [22] employed the traditional multilayer perceptron (MLP) technique to forecast the RUL of bearings during laboratory testing, demonstrating superior predictive performance compared to reliability-based alternatives. Fink *et al.* [23] presented a multi-layer neural network approach employing multi-valued neurons specifically designed to address the challenge of forecasting the performance and degradation time series. A case study was carried out with a specific focus on consider the deterioration of a railway turnout system. Khawaja *et al.* [24] devised a neural network method for predicting confidence that includes a confidence distribution node, addressing the limitation in neural network techniques where obtaining explicit confidence limits for RUL predictions proves challenging. Additionally, several fuzzy logic approaches have been integrated to MLP networks to enhance learning acquisition for PHM. Using RNN, Malhi *et al.* [25] proposed employing RNNs and competitive learning techniques for long-term prognostics regarding the health status of machinery. They utilized the continuous wavelet (WT) to preparation vibration singles obtained from a faulty rolling bearing, subsequently employing these preprocessed indicators as inputs for their model. The authors in the authors recommended an long short-term memory (LSTM) approach for RUL prediction in aero engines. This method was proposed to address scenarios involving highly intricate operations, hybrid faults, and substantial noise levels, thereby enhancing the capabilities beyond those offered by the norm RNN. Zhao *et al.* [26] applied LSTM networks to a tool wear health monitoring task. They integrated both frequency and time domain functions within their approach, Ren *et al.* [20] introduced an optimized DL technique designed for collaborative estimation of RUL in multiple bearings. They substantiated the method's viability and superiority through numerical evaluations conducted on a real dataset. Liao *et al.* [27] introduced an innovative restricted Boltzmann machine designed for representation learning aimed at determining the RUL of machines. This approach incorporates a novel regularization term along with an unsupervised self-organizing map algorithm. The study from Zhang *et al.* [28] presented a multi-objective DBN ensemble approach. This method combined one of the evolutionary algorithms with a conventional DBN training approach to concurrently develop multiple DBNs, emphasizing both accuracy and diversity in their construction.

In another study from Zheng *et al.* [29], the C-MAPPS benchmark dataset was used to predict the RUL of the turbofan engine using LSTM based on on-time sequence representation. The use of CNN to estimate the RUL of the same engine was proposed in [20]. The process uses a time window method as input feature to the suggested model. Hence, more degradation data should be collected. As a result, the dimension of model inputs has increased, causing difficulty in the development of the DNN model, that is, how to set up network nodes and network layers to avoid overfitting and reduce time and computational expenses while also avoiding getting stuck in local minimum points. Muneer *et al.* [30] provide four data-driven prognostic models that employ DNNs with an attention mechanism to precisely estimate the turbofan engines' RUL. Without requiring a prior understanding of prognostics or signal processing, the models increase DNN feature extraction by utilizing a sliding time window method. To enhance the prediction of RUL for turbofan engines, Muneer *et al.* [31] also provide a novel attention-based deep CNN design. The suggested model makes use of multivariate temporal information by selecting features based on the processability metric and preparing samples using a time window technique. Another study recently conducted by Peng *et al.* [1]. As a technique for RUL prediction, the combination of 1-D CNNs with LSTM and full convolutional layer (1-FCLCNN) was proposed. This technique extracts the spatial and temporal characteristics from the FD003 and FD001 datasets produced by the turbofan engine using LSTM and 1-FCLCNN. Researchers have also focused a great deal of emphasis on CNN applications in RUL-related disciplines [16]. Babu *et al.* [19] the deep CNN method was initially applied for RUL prediction. CNN fared better than the MLP, SVM, and SVR models, according to the data. The CNN method, which was suggested by [19] was examined and tested using the C-MAPSS dataset, yielding an RMSE of 18.45.

Similarly, Li *et al.* [10] suggested a deep CNN time window method for improved signal extraction. The method was tested on NASA's turbofan engine (C-MAPSS dataset) degradation problem and demonstrated a significant advantage. Even with the CNN model's high performance, additional optimization is still needed because it still takes longer to train than other shallow approaches. Furthermore, the recommended method has a heavy computational load. Wen *et al.* [32] created a brand-new residual CNN (ResCNN). ResCNN makes use of the residual block, which can help solve the vanishing/exploding gradient problem by using shortcut connections to bypass several convolutional layer blocks. Moreover, the k-fold ensemble method helped to enhance ResCNN. NASA's C-MAPSS benchmark dataset was used to test the suggested ensemble ResCNN. A new technique for deep features learning for RUL predictions utilizing multi-scale CNN (MS-CNN) and time-frequency representation (TFR) has been provided in another work suggested by [33]. The bearing deterioration signal's non-stationary character can be efficiently shown by TFR. By using WT, we were able to accumulate time series deterioration signals and create TFRs that are rich in valuable information. These TFRs were high dimensional, thus bilinear interpolation was utilized to reduce their size before they were utilized as inputs for the DL models. Nevertheless, the suggested method [33] exhibits a few limitations. Initially, the algorithm's training duration is sluggish, necessitating an enhancement in computational speed. Secondly, utilizing a graphical processing unit becomes imperative to assist in handling the primary TFR processing.

Additionally, Li *et al.* [34] aimed to enhance machines' RUL estimation by introducing a network structured as a controlled acyclic graph that merges LSTM and CNN to predict RUL. Li *et al.* [34] observed that when employing a singular timestamp as input, padding signals within the same training batch adversely impacted the overall predictive capability of the integrated approach. To mitigate this issue, the authors adopted their proposed method to create a short-term sequence, moving the time window (TW) in increments of duration single-phase. Additionally, they replaced the conventional linear function, based on the degradation mechanism, with a piece-wise RUL technique. In conclusion, the authors affirmed that augmenting the length of the time window could enhance the accuracy of their proposed model. In another study conducted by Zhang *et al.* [35] they employed CNN-based extreming gradient boosting (CNN-XGB) utilizing an extended of TW. This approach aimed to address challenges within aero-engine systems that often function across diverse operating conditions. These variations might impact the system's degradation path differently, potentially hindering the accuracy of RUL prediction. The suggested method underwent validation utilizing NASA C-MAPSS turbofan aero-engine datasets. It resulted in an RMSE of 20.3, with a reported training duration of 621.7 seconds. Wang *et al.* [36] proposed the MS-CNN to estimate the RUL of rolling bearing. The suggested approach by Liu *et al.* [36] aims to overcome the capability to learn local and global features synchronously limited to conventional CNN. Convolution filters with varying dilution rates were combined to form a dilated convolution block capable of learning features in a variety of receptive fields. Concatenating numerous stacked, integrated, and dilated convolution blocks varied depths allowed for the extraction of local and global features. The proposed method's effectiveness was validated by a benchmark dataset bearing called PRONOSTIA. Hence, in this study, we aimed to investigate different DNN models for RUL estimation to determine the technique with an excellent feature extraction and high capability to expect the RUL of a turbofan engine.

3. MATERIALS AND METHODS

This research utilizes a comparative analysis method to assess how effectively four prominent deep learning models can predict the RUL of various engines units. The proposed DNN-based models are rigorously trained and evaluated using well-known performance metrics. The initial section of the proposed methodology is dedicated to describing the four candidate deep learning models, while the subsequent sections outline the final two stages of the methodology.

3.1. Candidate model training and optimization

This part offers an in-depth overview of the DNN structures and optimization strategies implemented for creating candidate models to predict the RUL of turbofan engines. To achieve this goal, several commonly used NN architectures, including CNNs, RNNs, gated recurrent unit (GRU), and LSTM, were employed in this study. In addition, we applied the randomized hyperparameter search method, similar to that described in [34], to enhance the performance of the DNN models. This approach involves conducting a random search across a broad hyperparameter space, allowing for the identification of optimal hyperparameters with limited computational efforts. Specifically, we randomly sampled hyperparameters, created models using these parameters, and evaluated their performance. Subsequent subsections will provide concise descriptions of each DNN architecture used in this study for the RUL prediction of turbofan engines.

3.1.1. RNNs

Traditional DNNs have a limitation in that the individual neuron weights cannot identify exact representations of features for the corresponding RUL due to the complex system structure. To overcome this limitation, RNNs address this issue by incorporating a loop mechanism that operates over time steps. Specifically, a sequence vector $\{x_1, \ldots, x_n\}$ through a recurrence formula $r_t = f_\alpha(r_{t-1}, x_t)$, where f represents the activation function, α represents a set of parameters used at each time step t, and x_t is the input at timestep t [37]. This research explores three types of recurrent neurons for developing candidate RNN-based models: a basic RNN unit, GRU, and LSTM unit. The parameters controlling the connections between the hidden layers and input, as well as the connections between activations starting from the hidden layer and extending to the output layer, remain constant throughout each time step in a vanilla recurrent neuron. The operation of a fundamental recurrent neuron during the forward passing can be formulated in a specific manner, which will be elaborated on in the subsequent sections.

$$a^t = g(W_a[a^{<t-1>}, X^t] + b_a) \tag{1}$$

$$y^t = f(W_y a^t + b_y) \tag{2}$$

At each timestep t, the activation function g is denoted by g, where t represents the current timestep and X^t represents the input at that timestep. The bias is represented by b_a, and W_a represents the cumulative weights at timestep t for the activation output denoted by a^t. The output of the activation, a^t, can be utilized to generate forecasts for y_t at time t, if required.

The model employs an embedding layer to map the RUL into a vector space of dimension R20, transforming semantic relationships into geometric ones. The successive layers of the DNN examine these geometric shapes in order to identify and understand complex feature representations, which are then evaluated by the output layer to make predictions, employing a singular sigmoid unit. Despite the effectiveness of DNNs using basic RNN neurons in several domains, these models encounter challenges pertaining to the vanishing gradient problem and their limited capacity to capture long-term relationships. In order to address these obstacles, the scholarly community has suggested alternate designs for recurrent neurons, namely the GRU [38] and the LSTM [39], which have shown improved performance in mitigating the vanishing gradients problem and aiding the acquisition of long-term dependence [40].

Nascer *et al.* [41] presented a GRU model that demonstrates enhanced efficacy in the task of long-term relationship learning within time-series datasets. The operational characteristics of the GRU may be mathematically described using the following set of equations:

$$\overline{H}^t = \tanh(W_c[\Gamma_r * H^t, X^t] + b_c) \tag{3}$$

$$\Gamma_r = \sigma(W_r[H^{(t-1)}, X^t] + b_r) \tag{4}$$

$$\Gamma_u = \sigma(W_u[H^{(t-1)}, X^t] + b_u) \tag{5}$$

$$H^t = \Gamma_u \cdot \overline{H}^t + (1 - \Gamma_u) \cdot H^{(t-1)} \tag{6}$$

$$a^t = H^t \tag{7}$$

W_r, W_c, and W_u are the weight matrices, while b_r, b_c, and b_u are the bias terms for the input X_t at each time step t. σ represents the logistic regression function, and a^t is the activation value at time step t. Except for GRU neurons, the RNN model employing GRU is similar to those using plain RNN neurons. Table 2 shows the GRU-based RNN model architecture for RUL estimation. Hochreiter and Schmidhuber [39] added the LSTM neuron, which improved on the RNN unit and made the GRU more robust. The following differences between GRU and LSTM cells:

- In standard LSTM units, there is no significant gate like Γ_r used in the computation of $\overline{H}^t$.
- LSTM units utilize two distinct gates, namely the output gate Γ_o and the update gate Γ_u, instead of just relying on an update gate Γ_u. The activation outputs of the LSTM unit for other hidden units in the network are computed by the output gate, which monitors the visibility of the content in the memory cell (H^t). In contrast, the update gate regulates the extent of information replacement on the previous hidden state, $H^{(t-1)}$, in order to get the updated hidden state, H^t. This process entails the determination of the extent to which information stored in memory cells should be discarded in order to guarantee optimal functionality.
- LSTM units employ two apparent gates in place of a single update gate Γ_u found in GRU units. These are the output gate Γ_o and the forget gate Γ_u. The output gate is responsible for regulating the visibility of the memory cell content H^t in calculating the activation outputs of the LSTM unit for other hidden units in the network. The forget gate, on the other hand, manages the degree to which the previous memory content $H^{(t-1)}$ is overwritten to produce H^t. This involves determining the extent to which information in the memory cell should be disregarded to maintain effective functioning.
- A key difference between LSTM and GRU architectures is that in LSTM, the content of the memory cell H^t might not be the same as the activation value a^t at time t.

Furthermore, the LSTM model which is based on the RNN method was developed with an architectural design that has a strong resemblance to both the GRU and basic RNN models. The only differentiation exists in the use of LSTM units inside the recurrent layers. Figure 1 is structure of a deep RNN-based model proposed for RUL estimation.

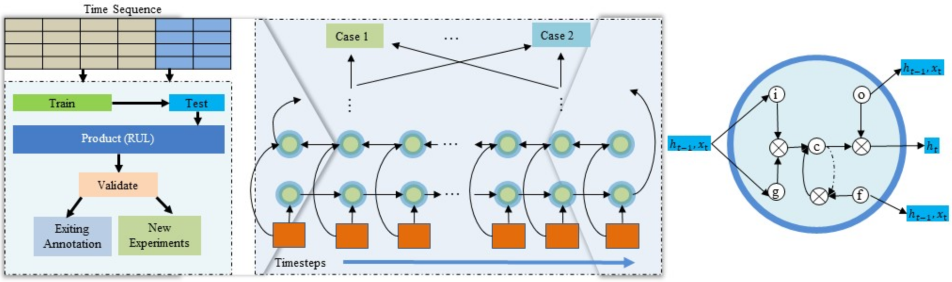


Figure 1. Structure of a deep RNN-based model proposed for RUL estimation

3.1.2. CNNs

CNNs are particularly effective at processing learning tasks that entail complex spatial patterns in high-dimensional input data. Such challenges are prevalent in various domains, including, but not limited to, image processing [42], video analysis [43], analysis of amino acid sequences [41], [44] and the examination of time-series failure signals. The primary objective of CNNs is to learn hierarchical filters capable of transforming large input data into precise class labels while employing a minimal number of trainable parameters. This transformation is accomplished through sparse interactions between the input data and trainable parameters, facilitated by a mechanism known as parameter sharing. This method allows CNNs to acquire representations that are equivariant, also known as feature maps, of the intricate and spatially organized input data [45]. In a deep CNN, the units in the deep layers have the ability to indirectly interact with a significant percentage of the input data. This is achieved through the use of pooling operations. Pooling operations streamline the output at a particular point by employing a statistical summary, enabling the network of the model to acquire intricate properties from this compacted representation map [10]. The topmost section of the CNN typically includes many fully connected layers (FCL), including the output layer, leveraging the intricate information acquired by the preceding layers to make predictions.

The architecture based on CNN that is used for the RUL prediction, which consists of two convolution-maxpool blocks in the embedding layer, a global average layer, and an output layer of the sigmoid neuron. The learning efficiency of the CNN model is significantly improved through the use of multiple non-linear feature extractions. This enables the model to autonomously learn hierarchical data representations. Consequently, the size of the convolution kernel and the quantity of convolution layers greatly influence the model's predictive capabilities. Figure 2 illustrates the CNN architecture designed for RUL estimation in this study. The initial input data are in a two-dimensional (2D) format, where one dimension represents the feature number in a 1D format, and the other dimension corresponds to the sensor's time sequence, also in 1D.

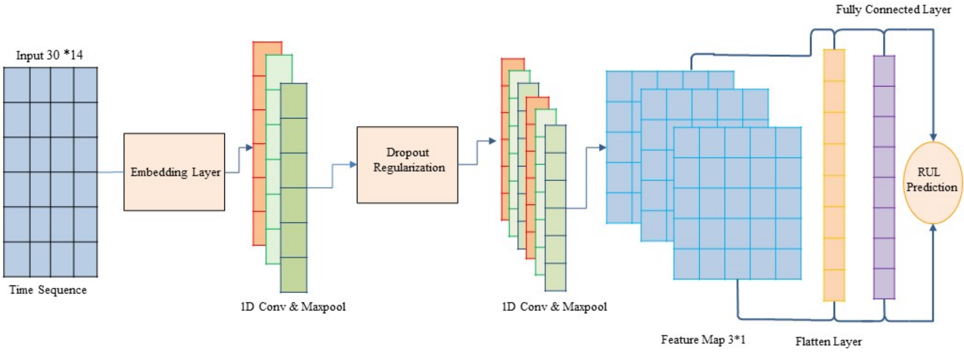


Figure 2. The proposed architecture for RUL prediction using a deep CNN

After that, the CNN model processes the input data through four convolutional layers, each with a similar structure, to extract features. These extracted features are then integrated with a convolutional layer equipped with a single filter, sized 3 × 1. After the feature maps are flattened, they are connected to a fully connected layer. To mitigate overfitting, a dropout method is applied. The activation function for each layer is the ReLU. In this research, the optimization of the model is handled by the Stochastic Gradient Descent (SGD) algorithm. Considering the current characteristics of the turbofan aeroengine datasets, our models has been adjusted to impose a higher penalty for delayed (lag) predictions. The formulation of this loss is specified as follows in the study.

$$\text{loss} = \frac{1}{N}\sum_{i=1}^{N}\omega(y_i - \hat{y}_i)^2 \tag{8}$$

Where y_i is the actual value and $\hat{y}_i$ is the predicted value. N is the validation set sample count. Penalty coefficient ω is set to 1 if real value y_i exceeds anticipated value $\hat{y}_i$, and to 2 if actual value is less than expected value.

3.2. Data pre-processing and normalization

In practical scenarios, raw data from sensors, operational parameters, and run-to-failure information are typically accessible. To prepare the data for training and testing, it is necessary to standardize the values of each sensor, as the scales may be different. In the experiment conducted, data from 21 sensors were utilized, and any anomalous or unvarying data was excluded. The normalization technique used is Min-Max scaler, was applied to each feature to scale the data into a range between 0 and 1. In addition, for systems where the health decay is not linear from the beginning of operations, piece-wise functions can be used to enhance the precision of the estimated $\text{RUL}^t_{\text{calc}}$. Also, if information about varying workloads, operational environments, and specific modes of deterioration is available, it can be integrated into the RUL estimation model to further refine its accuracy in certain applications. Figure 3 explains the measurement and the raw input of FD001 dataset for RUL of each sensor.

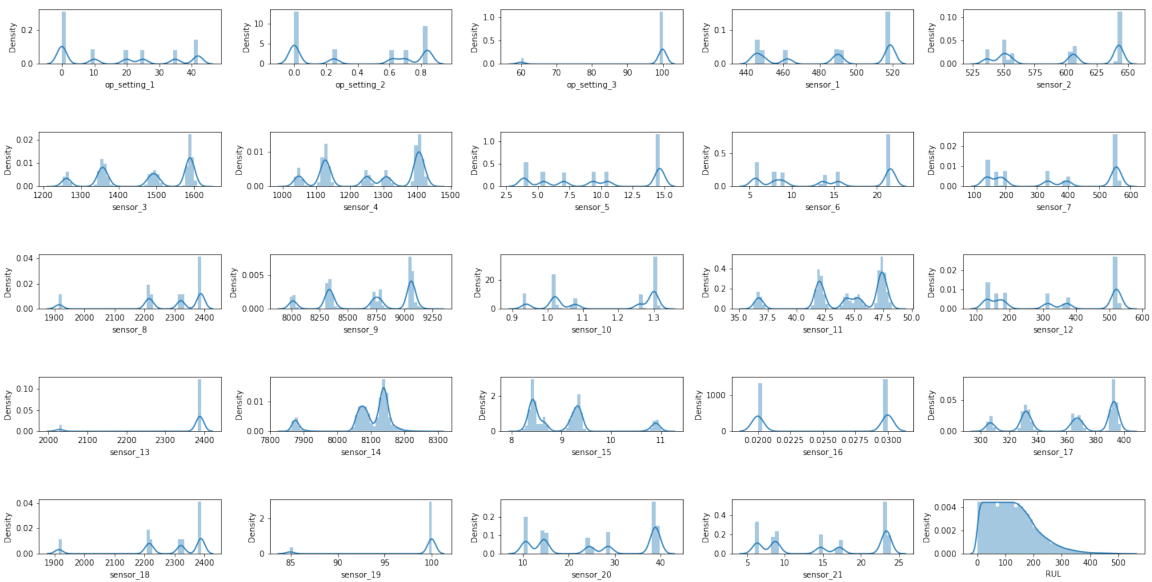


Figure 3. RUL display plot for each sensor measurement and the raw input of FD001 dataset

The second method is Min-Max normalization, which involves scaling the raw data from the sensors to fit within the range of 0 and 1. To achieve this, the sensor's minimum and maximum readings data are identified, and these values are used to map the data onto the range -1 and 1. The normalized sensor output $\hat{x}_i$ is calculated by taking the ratio of the difference between the original sensor output x_i and the minimum value, to the range, which is the difference between the maximum and minimum values. It is important to note that normalization is necessary because different sensors may have different value scales, and normalizing the data allows for fair comparison and accurate training and testing of the models. Furthermore, in certain applications, such as those with non-linear RUL decay, piece-wise functions can be used to adjust the estimated $\text{RUL}_{\text{calc}}^{t}$ goals. Incorporating knowledge of different workloads, operational environments, and deterioration modes into the RUL estimation model can also improve its accuracy if such information is available.

$$\mathbf{x}_i' = \frac{\mathbf{x}_i - \min \mathbf{x}_i}{\max \mathbf{x}_i - \min \mathbf{x}_i} \tag{9}$$

Additionally, to incorporate multivariable temporal information, a time window (TW) approach is adopted, as previously done in a study [10]. For the training dataset FD001, a TW length of 30 was selected, and all historical data within the TW was extracted to form a high-dimensional input vector. This vector has a length of 14 × 30, using 14 out of the 21 available sensors as raw input features. In this study, the developed DNN-based models were specifically intended to forecast the RUL of aero-engines operating under a single condition. Consequently, the FD001 dataset, which comprises data collected under a single operating condition, was chosen for experimental analysis. The structure of the network used for feature extraction was adapted to align with the dynamic qualities of the operational data of an aero-engine, which can vary across different operating conditions.

3.3. Performance metrics

In this study, prognostic performance was assessed using three metrics: R-squared (R^2), mean absolute error (MAE), and RMSE. The rationale behind the selection of these three indicators is their extensive application in cutting-edge model performance assessment. The first evaluation metric, RMSE, is presented:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} d_i^2} \tag{10}$$

MAE is the sum of anticipated errors or the mean of all absolute errors:

$$\text{MAE} = \frac{1}{n}\sum_{n} |X_P - X| \tag{11}$$

Thus, X_P is estimated data, X is the ground truth data, and n is the number of samples. Statistical measure is R^2 shows how much of the variation of a dependent variable can be accounted for by an independent variable:

$$R^2 = 1 - \frac{RSS}{TSS} \tag{12}$$

where TSS is the total sum of squares, RSS is the sum of residual squares, and R^2 is the determination coefficient.

3.3.1. Prognostic procedure

Figure 4 shows the multi-phase prognostic experimental strategy. Preprocessing began with the extraction of 14 raw sensor values and normalization to scale the FD001 dataset inside the [-1, 1] range. We then produced training and testing datasets with time sequence information limited to Ntw. DNN models used pre-provided 2D standardized data. It was unnecessary to manually construct signal processing features like skewness and kurtosis. Thus, no prognostics or signal processing knowledge is required. This was followed by building the proposed deep neural network models for life RUL prediction and specifying their hidden layer count, convolution filter size, and other parameters. The DNN models were trained using normalized training data and labeled RUL values for training samples. Back-propagation learning and mini-batches in SGD updated the network's weight. To train each epoch, the data were randomly divided into several tiny batches of 512 samples. Use the micro batch mean loss function to tweak each layer's weights in the training deep neural network model. Experimental experiments determined the best batch size of 512 samples, which was employed in all case studies. To assure convergence, a variable learning rate was used, starting at 0.005 for the first 25 optimization epochs and then progressing to 0.001. DNN candidate models cannot exceed 250 training epochs by default.

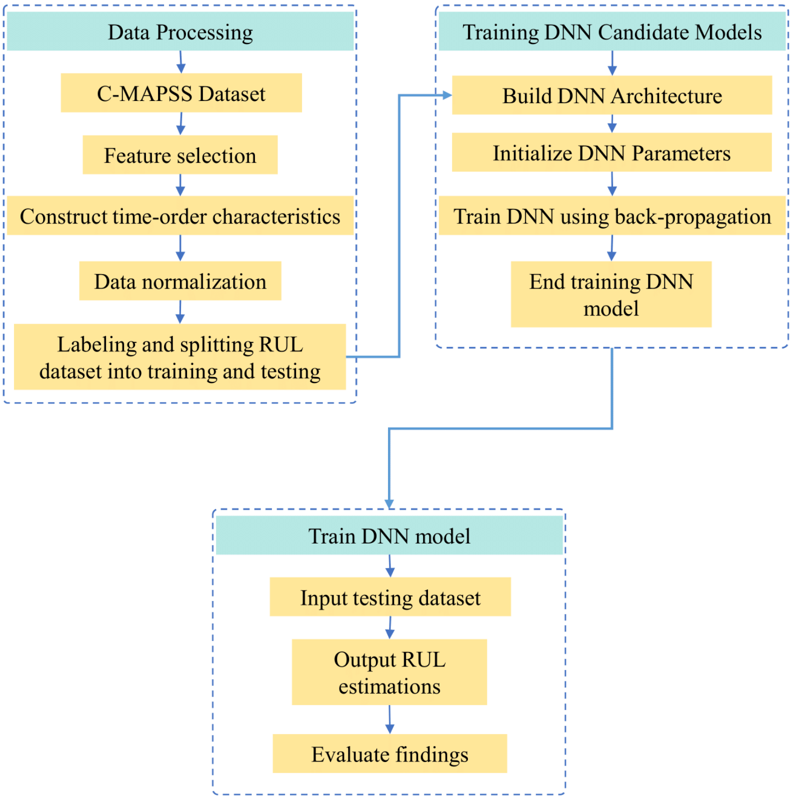


Figure 4. Prediction process of our proposed approach

4. EXPERIMENTAL RESULTS

This section presents a summary of the experimental findings and discusses their significance. Firstly, the C-MAPSS benchmark dataset is introduced in the first subsection. Secondly, the experimental results and performance analysis are presented in the second subsection. Finally, the last subsection provides a comparative analysis with existing literature.

4.1. Benchmark dataset for C-MAPSS

The C-MAPSS dataset serves as a widely utilized resource in advanced prognostic research, comprising four sub-datasets that depict the engine's behavior under diverse operational conditions and mechanisms of failure [46]. Each subset includes both training and testing sets, accompanied by actual RUL values. These subsets are characterized by 21 sensors and three operational settings [47]. Each engine unit undergoes distinct levels of deterioration, gradually degrading over time until it reaches a point of system failure, marking the culmination of an unhealthy operational cycle. As a result, sensor recordings in the testing set cease before the occurrence of the system fault. The dataset is presented in a compressed text format, where individual rows signify data snapshots taken within a single operational cycle, and each column corresponds to a distinct variable. Table1 1 provide comprehensive details about the dataset. The objective of the experiment was to predict the RUL of the engine unit in the testing set and that of a single-engine unit. For the purposes of this research, only the first subset of data labeled FD001 was utilized for the verification of the DNN models. Consequently, this data subset consisted of 100 training samples and 100 test samples.

Table 1. Description of C-MAPSS benchmark dataset

C-MAPSS Dataset	FD001
Engine units for training	100
Engine units for testing	100
Operating conditions	1
Fault modes	1

4.2. Analyses of candidate model performance and experimental results for 100 testing engines

This subsection discusses the prognostic performance of the suggested DNN-based models for RUL estimation. An analysis was conducted to investigate the effects of various factors on the outcomes, such as the quantity of concealed layers and residual scatter plots for each model. The comparison of the deep structure of the proposed four models with that of other prominent NN architectures demonstrated the proposed DNN-based models' effectiveness. Additionally, the proposed approach's superiority was proven by comparing the most recent state-of-the-art prognostic outcomes on the same C-MAPSS dataset. Figure 5 shows the RNN-based model prediction for 100 engine units in the FD001 dataset. The graph's X-axis represents the actual RUL values, where the Y-axis of the graph denotes the predicted RUL values across the whole testing dataset. Figure 6 shows FD001 test dataset residual analysis of an LSTM-based model (best model).

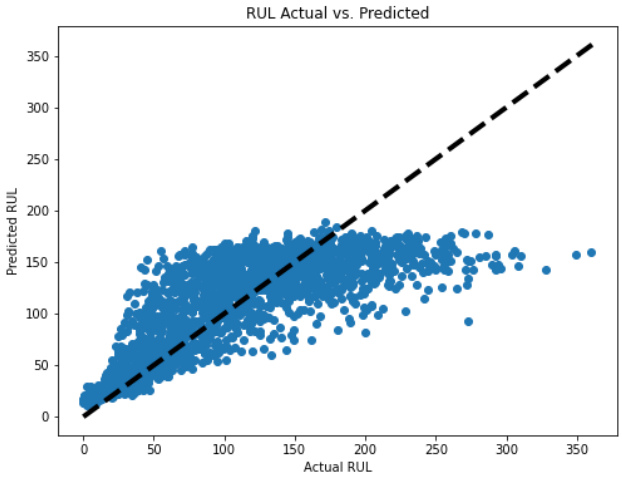


Figure 5. Sorting predication for the 100 testing engine units in FD001 using the LSTM-based model

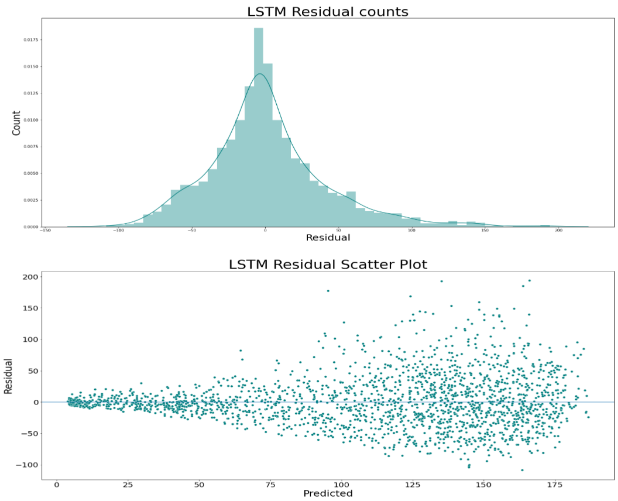


Figure 6. FD001 test dataset residual analysis of an LSTM-based model (best model)

5. RESULT AND ANALYSIS

After analysing the evaluation metrics, it becomes evident that the decision tree classifier (DTC) and XGBoost Classifier (XGBC) models display the most elevated accuracy scores in comparison to the other models. Nonetheless, when scrutinizing the precision and recall scores, it is clear that the DTC exhibits the lowest values. This means that the DTC is more likely than the other models to produce a larger number of false positives and a lower number of true positives. Conversely, the XGBC model showcases the highest precision and recall scores, indicating that it has a lower probability of producing false positives and a higher probability of generating true positives than the other models. Residual analysis is a popular assessment measure in regression issues that tries to assess the goodness of fit of a model by establishing residuals and studying residual plots. A residual (e) is precisely the discrepancy between the observed and predicted values of a dependent variable (y) and its corresponding estimated value $(\hat{y}_i)$. The residual (e_i) for the $(i)^t h$ data point may be expressed as follows:

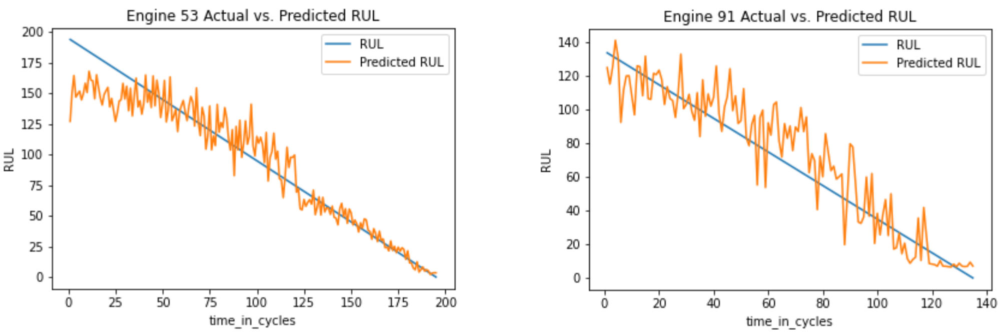
$$e_i = \bar{y}_i - y_i \tag{13}$$

The evaluation of a regression model can be conducted through residual analysis, a method that assesses the model's suitability by investigating the plots of residual. A residual (e) represents the disparity between the dependent variable's measured value (y_i) and its predicted value $(\hat{y}_i)$ for a specific data point (i). Residual analysis is instrumental in identifying whether the model tends to under- or over-estimate. Over-estimation occurs when $e_i > 0$, indicating consistent over-prediction $(\hat{y}_i)$ compared to actual scores (y_i). Conversely, under-estimation is indicated by $e\ _i <$, signifying persistent predictions of $(\hat{y}_i)$ being lower than (y_i).

Residual plots unveil valuable insights into model behavior by revealing patterns of under- or over-estimation. An ideal regression model achieves a balance between over- and under-estimation, resulting in a symmetric distribution of residuals (ei) around 0, with approximately equal instances above and below zero. These insights are graphically presented using scatter plots and distribution plots, illustrating the distribution of (ei) around 0. The extent of skewness in the residual distribution indicates the model's bias, with a more skewed distribution suggesting greater bias, and conversely. Figures 6 display the residual analysis of the LSTM-based model for the FD001 test data.

5.1. Analysis of experimental results and the performance of candidate models for single-engine prediction

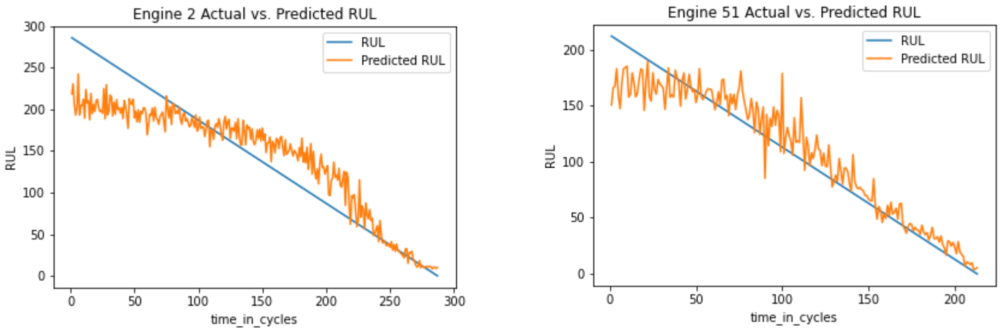
This section assesses and examines several DNN models for predicting the RUL of turbofan engines. The actual and predicted RUL values of the testing units are compared to enhance the analysis of prognostic performance. This comparison is illustrated in Figure 7, where the testing units are arranged in ascending order based on their actual RUL. Figure 7 also presents the RUL prediction results obtained from the LSTM architecture in the initial validation test, including two randomly selected examples. Figure 7(a) shows the RUL estimation for Engine #53, while Figure 7(b) illustrates the RUL prediction for Engine #91. Furthermore, the predicted RUL values generated by the proposed LSTM architecture demonstrate a close agreement with the actual RUL for two randomly selected engines, as further illustrated in Figure 8.



(a) (b)

Figure 7. Displays the results of the RUL forecast obtained from the LSTM architecture in the initial verification test; (a) depicts the RUL estimate for the first randomly chosen example, Engine #53 and (b) exhibits the RUL prognosis for the second randomly chosen example, Engine #91

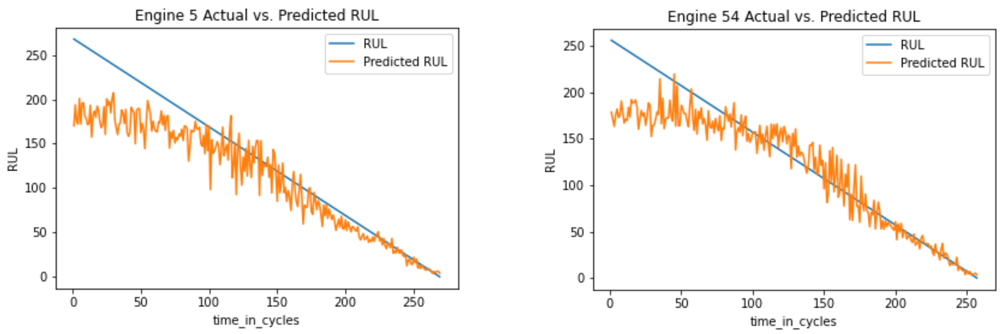
Figure 8 presents the RUL prediction results obtained using the proposed GRU architecture in the second validation test. The comparison between actual and predicted RUL values demonstrates the effectiveness of the GRU model in capturing degradation trends. In particular, Figure 8(a) illustrates the RUL forecast for the first randomly selected example, Engine #2, while Figure 8(b) depicts the RUL forecast for the second randomly selected example, Engine #51.



(a) (b)

Figure 8. Presents the RUL prediction results obtained through the proposed GRU architecture in the second validation test; (a) illustrates the RUL forecast for the first randomly chosen example (Engine #2) and (b) depicts the RUL forecast for the second randomly chosen example (Engine #51)

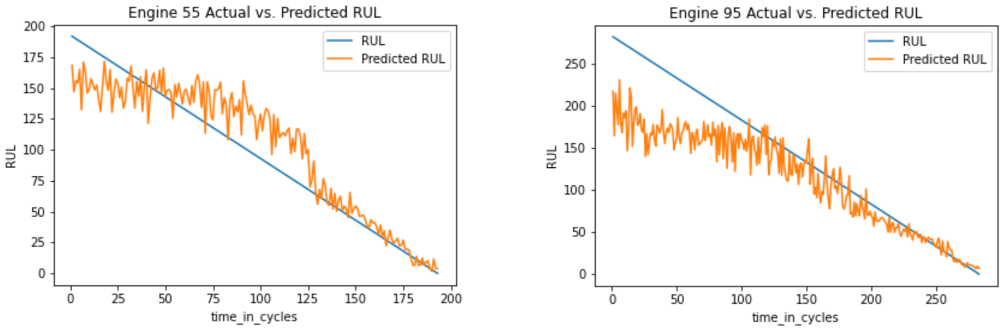
The second verification test, which employed a candidate model based on a GRU, demonstrated the model's capacity to accurately predict the RUL for two randomly selected instances (engines 2 and 52), as shown in Figure 9. The third verification test utilized a S-RNN candidate model architecture. The results showed that the RUL estimation was less precise compared to the other two RNN techniques, specifically GRU and LSTM. In this figure, Figure 9(a) displays the RUL forecast for the first randomly selected example, Engine #5, while Figure 9(b) illustrates the RUL forecast for the second randomly selected example, Engine #54.



(a) (b)

Figure 9. The RUL prediction outcomes utilizing the proposed S-RNN architecture in the third verification test are presented (a) displays the RUL forecast for the first randomly chosen example (Engine #5) and (b) illustrates the RUL forecast for the second randomly chosen example (Engine #54)

Ultimately, the fourth verification test employed the suggested CNN candidate model, showcasing the predicted RUL findings in Figure 10. The CNN-based model had relatively inferior performance in both instances. Therefore, in this work, the LSTM-based model exhibited greater performance in comparison to the CNN, S-RNN, and GRU models in both evaluated scenarios. In this figure, Figure 10(a) shows the estimated RUL for the first randomly selected example, Engine #55, while Figure 10(b) illustrates the forecasted RUL for the second randomly selected instance, Engine #95.



(a) (b)

Figure 10. Displays the results of the RUL forecast generated from the proposed CNN architecture during the fourth verification test; (a) shows the estimated RUL for the first randomly chosen example, Engine #55 and (b) illustrates the forecasted RUL for the second randomly chosen instance, Engine #95

5.2. Comparison with literature

In this section, a comparative analysis is presented between the proposed four Deep Neural Network (DNN)-based predictors and state-of-the-art techniques for RUL prediction. Prior research has utilized diverse DL techniques to predict RUL using the C-MAPSS benchmark dataset. Table 2 summarizes the comparison of the proposed deep candidate models with contributions from the current literature. While the table exclusively presents available metrics, it underscores the promising outcomes achieved by the proposed DNN-based predictor. The experimental findings affirm that the proposed DNN-based predictors exhibit superior performance compared to other methods previously employed for predicting RUL on the independent testing dataset of the benchmark. It is noteworthy that, with the exception of a single study reporting better results, the proposed DNN-based predictors consistently outperform prior approaches. Table 2 presents a comparison between our proposed GRU-based and LSTM-based models and the previous models reported in the literature. The experimental results show that our proposed models outperform all the previous models. The only study that achieved better performance is presented by [10], who used five convolution layers with different filter sizes. However, the network structure of our proposed DNN model is different. Through the combination of the Time-Warping (TW) approach and temporal information of signals, the LSTM-based predictor demonstrates proficiency in capturing long-range relationships and extracting characteristics from the time-frequency domain.

Table 2. Compares the proposed DNNs-based approach to comparable literature contributions

Prediction model	FD001		
	MAE	RMSE	R2
Proposed CNN Predictor	8.54	21.88	0.2852
Proposed SRNN Predictor	5.45	17.13	0.4201
Proposed GRU Predictor	2.18	16.32	0.4350
Proposed LSTM Predictor	2.03	15.30	0.4354
Echo State Network with Kalman Filter [48]	Not Reported	63.46	Not Reported
SVM [48]	Not Reported	29.82	Not Reported
MLP [19]	Not Reported	37.56	Not Reported
Deep CNN [19]	Not Reported	18.45	Not Reported
DW-RNN [49]	Not Reported	22.52	Not Reported
MTL-RNN [49]	Not Reported	21.47	Not Reported

Experimental results indicate that an augmentation in the sliding TW corresponds to enhanced accuracy in predicting RUL. The proposed improved LSTM-based model exhibits precise RUL predictions for aero-engines without necessitating an in-depth understanding of engine construction, failure mechanisms, or specialized knowledge and expertise. This study streamlines the modeling process and serves as a practical decision-making tool for aircraft engine maintenance and health management.

6. CONCLUSION

To conclude, achieving effective predictive maintenance and preventing catastrophic failures and casualties can be accomplished through the use of a reliable and accurate estimate of RUL. Furthermore, the increasing adoption of intelligent manufacturing has led to the increased interest in data-driven RUL techniques among academic and engineering communities. Four distinct DNN models were developed in this study to accurately predict the RUL of a turbofan engine. The models were trained and evaluated using the C-MAPSS benchmark dataset. Precisely forecasting the RUL of aero-engines is crucial for enhancing the dependability and security of aero-engine systems. The LSTM-based and GRU-based learning models exhibit superior prediction accuracy compared to the other models when their results are compared. Moreover, as the level of deterioration intensifies, the forecast outcomes become increasingly precise. Although the suggested method has shown encouraging experimental outcomes, there is potential for additional enhancement of the structure in future research. More precisely, the suggested method will be expanded to encompass the forecasting of RUL for additional subsets of data and for turbofan engines running in varying situations. As the operating circumstances get more complicated, predicting the RUL becomes increasingly problematic, therefore requiring additional analysis.

FUNDING INFORMATION

The study mentioned in this paper is funded by Universiti Teknologi PETRONAS through the Yayasan Universiti Teknologi PETRONAS Fundamental Grant (015LC0-311) and Yayasan Universiti Teknologi PETRONAS Prototype Grant (015PBC-027).

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

Data availability is not applicable to this paper as no new data were created or analyzed in this study.

REFERENCES

[1] C. Peng, Y. Chen, Q. Chen, Z. Tang, L. Li, and W. Gui, "A remaining useful life prognosis of turbofan engine using temporal and spatial feature fusion," *Sensors*, vol. 21, no. 2, p. 418, 2021.

[2] A. Muneer, S. M. Taib, S. M. Fati, A. O. Balogun, and I. A. Aziz, "A hybrid deep learning-based unsupervised anomaly detection in high dimensional data," *Computers, Materials & Continua*, vol. 70, no. 3, 2022.

[3] J. Wei, P. Bai, D. Qin, T. C. Lim, P. Yang, and H. Zhang, "Study on vibration characteristics of fan shaft of geared turbofan engine with sudden imbalance caused by blade off," *Journal of Vibration and Acoustics*, vol. 140, no. 4, p. 041010, 2018.

[4] P. Zschech, J. Bernien, and K. Heinrich, "Towards a taxonomic benchmarking framework for predictive maintenance: The case of nasa's turbofan degradation," in *ICIS*, 2019.

[5] H. Tuzcu, Y. Sohret, and H. Caliskan, "Energy, environment and enviroeconomic analyses and assessments of the turbofan engine used in aviation industry," *Environmental Progress & Sustainable Energy*, vol. 40, no. 3, p. e13547, 2021.

[6] Y. You, J. Sun, B. Ge, D. Zhao, and J. Jiang, "A data-driven m2 approach for evidential network structure learning," *Knowledge-Based Systems*, vol. 187, p. 104810, 2020.

[7] F. N. B. Sohaidan, A. Muneer, and S. M. Taib, "Remaining useful life prediction of turbofan engine using long-short term memory," in *2021 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT)*. IEEE, 2021, pp. 1–6.

[8] P. R. D. O. Da Costa, A. Akcay, Y. Zhang, and U. Kaymak, "Attention and long short-term memory network for remaining useful lifetime predictions of turbofan engine degradation," *International journal of prognostics and health management*, vol. 10, no. 4, 2019.

[9] S. Ghorbani and K. Salahshoor, "Estimating remaining useful life of turbofan engine using data-level fusion and feature-level fusion," *Journal of Failure Analysis and Prevention*, vol. 20, pp. 323–332, 2020.

[10] X. Li, Q. Ding, and J.-Q. Sun, "Remaining useful life estimation in prognostics using deep convolution neural networks," *Reliability Engineering & System Safety*, vol. 172, pp. 1–11, 2018.

[11] F. Elasha, S. Shanbr, X. Li, and D. Mba, "Prognosis of a wind turbine gearbox bearing using supervised machine learning," *Sensors*, vol. 19, no. 14, p. 3092, 2019.

[12] G. Hou, S. Xu, N. Zhou, L. Yang, and Q. Fu, "Remaining useful life estimation using deep convolutional generative adversarial networks based on an autoencoder scheme," *Computational Intelligence and Neuroscience*, vol. 2020, 2020.

[13] G. Zhao, S. Wu, and H. Rong, "A multi-source statistics data-driven method for remaining useful life prediction of aircraft engine," *Journal of Xi'an Jiaotong University*, vol. 51, no. 11, pp. 150–155, 2017.

[14] Z. Zhang, X. Si, C. Hu, and Y. Lei, "Degradation data analysis and remaining useful life estimation: A review on wiener-process-based methods," *European Journal of Operational Research*, vol. 271, no. 3, pp. 775–796, 2018.

[15] Z. Zhao, B. Liang, X. Wang, and W. Lu, "Remaining useful life prediction of aircraft engine based on degradation pattern learning," *Reliability Engineering & System Safety*, vol. 164, pp. 74–83, 2017.

[16] L. Wen, Y. Dong, and L. Gao, "A new ensemble residual convolutional neural network for remaining useful life estimation," *Math. Biosci. Eng*, vol. 16, no. 2, pp. 862–880, 2019.

[17] J. Wang, J. Zhang, and X. Wang, "A data driven cycle time prediction with feature selection in a semiconductor wafer fabrication system," *IEEE Transactions on Semiconductor Manufacturing*, vol. 31, no. 1, pp. 173–182, 2018.

[18] A. Z. Hinchi and M. Tkiouat, "Rolling element bearing remaining useful life estimation based on a convolutional long-short-term memory network," *Procedia Computer Science*, vol. 127, pp. 123–132, 2018.

[19] G. S. Babu, P. Zhao, and X.-L. Li, "Deep convolutional neural network based regression approach for estimation of remaining useful life," in *Database Systems for Advanced Applications: 21st International Conference, DASFAA 2016, Dallas, TX, USA, April 16-19, 2016, Proceedings, Part I 21*. Springer, 2016, pp. 214–228.

[20] L. Ren, Y. Sun, H. Wang, and L. Zhang, "Prediction of bearing remaining useful life with deep convolution neural network," *IEEE access*, vol. 6, pp. 13 041–13 049, 2018.

[21] X. Chen, S. Wang, B. Qiao, and Q. Chen, "Basic research on machinery fault diagnostics: Past, present, and future trends," *Frontiers of Mechanical Engineering*, vol. 13, pp. 264–291, 2018.

[22] A. Fentaye, A. Baheta, S. Gilani, and K. Kyprianidis, "A review on gas turbine gas-path diagnostics: State-of-the-art methods, challenges and opportunities," *Aerospace*, vol.6, no.7, 2019.

[23] O. Fink, E. Zio, and U. Weidmann, "Predicting component reliability and level of degradation with complex-valued neural networks," *Reliability Engineering & System Safety*, vol. 121, pp. 198–206, 2014.

[24] T. Khawaja, G. Vachtsevanos, and B. Wu, "Reasoning about uncertainty in prognosis: a confidence prediction neural network approach," in *NAFIPS 2005-2005 Annual Meeting of the North American Fuzzy Information Processing Society*. IEEE, 2005, pp. 7–12.

[25] A. Malhi, R. Yan, and R. X. Gao, "Prognosis of defect propagation based on recurrent neural networks," *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 3, pp. 703–711, 2011.

[26] R. Zhao, J. Wang, R. Yan, and K. Mao, "Machine health monitoring with lstm networks," in *2016 10th international conference on sensing technology (ICST)*. IEEE, 2016, pp. 1–6.

[27] L. Liao, W. Jin, and R. Pavel, "Enhanced restricted boltzmann machine with prognosability regularization for prognostics and health assessment," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 11, pp. 7076–7083, 2016.

[28] C. Zhang, P. Lim, A. K. Qin, and K. C. Tan, "Multiobjective deep belief networks ensemble for remaining useful life estimation in prognostics," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 10, pp. 2306–2318, 2016.

[29] S. Zheng, K. Ristovski, A. Farahat, and C. Gupta, "Long short-term memory network for remaining useful life estimation," in *2017 IEEE international conference on prognostics and health management (ICPHM)*. IEEE, 2017, pp. 88–95.

[30] A. Muneer, S. M. Taib, S. Naseer, R. F. Ali, and I. A. Aziz, "Data-driven deep learning-based attention mechanism for remaining useful life prediction: Case study application to turbofan engine analysis," *Electronics*, vol. 10, no. 20, p. 2453, 2021.

[31] A. Muneer, S. M. Taib, S. M. Fati, and H. Alhussian, "Deep-learning based prognosis approach for remaining useful life prediction of turbofan engine," *Symmetry*, vol. 13, no. 10, p. 1861, 2021.

[32] L. Wen, X. Li, L. Gao, and Y. Zhang, "A new convolutional neural network-based data-driven fault diagnosis method," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 7, pp. 5990–5998, 2017.

[33] J. Zhu, N. Chen, and W. Peng, "Estimation of bearing remaining useful life based on multiscale convolutional neural network," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 4, pp. 3208–3216, 2018.

[34] J. Li, X. Li, and D. He, "A directed acyclic graph network combined with CNN and LSTM for remaining useful life prediction," *IEEE Access*, vol. 7, pp. 75 464–75 475, 2019.

[35] X. Zhang *et al.*, "Remaining useful life estimation using cnn-xgb with extended time window," *IEEE Access*, vol. 7, pp. 154 386–154 397, 2019.

[36] H. Liu, R. Yuan, Y. Lv, H. Li, E. D. Gedikli, and G. Song, "Remaining useful life prediction of rolling bearings based on segmented relative phase space warping and particle filter," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–15, 2022.

[37] A. Muneer, A. Alwadain, M. G. Ragab, and A. Alqushaibi, "Cyberbullying detection on social media using stacking ensemble learning and enhanced BERT," *Information*, vol. 14, no. 8, p. 467, 2023.

[38] K. Cho, B. Van Merri¨enboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*, 2014.

[39] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[40] S. M. Al-Selwi, M. F. Hassan, S. J. Abdulkadir, and A. Muneer, "LSTM inefficiency in long-term dependencies regression problems," *Journal of Advanced Research in Applied Sciences and Engineering Technology*, vol. 30, no. 3, pp. 16–31, 2023.

[41] S. Naseer, R. F. Ali, A. Muneer, and S. M. Fati, "Iamidev-deep: Valine amidation site prediction in proteins using deep learning and pseudo amino acid compositions," *Symmetry*, vol. 13, no. 4, p. 560, 2021.

[42] V. Durairajah, S. Gobee, and A. Muneer, "Automatic vision based classification system using DNN and SVM classifiers," in *2018 3rd International Conference on Control, Robotics and Cybernetics (CRC)*. IEEE, 2018, pp. 6–14.

[43] A. Muneer and S. M. Fati, "Efficient and automated herbs classification approach based on shape and texture features using deep learning," *IEEE Access*, vol. 8, pp. 196 747–196 764, 2020.

[44] S. Naseer, R. F. Ali, S. M. Fati, and A. Muneer, "Computational identification of 4-carboxyglutamate sites to supplement physiological studies using deep learning," *Scientific Reports*, vol. 12, no. 1, p. 128, 2022.

[45] S. Naseer and Y. Saleem, "Enhanced network intrusion detection using deep convolutional neural networks," *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 12, no. 10, pp. 5159–5178, 2018.

[46] D. K. Frederick, J. A. DeCastro, and J. S. Litt, "User's guide for the commercial modular aero-propulsion system simulation (c-mapss)," *Tech. Rep.*, 2007.

[47] R. Huang, L. Xi, X. Li, C. R. Liu, H. Qiu, and J. Lee, "Residual life predictions for ball bearings based on self-organizing map and back propagation neural network methods," *Mechanical systems and signal processing*, vol. 21, no. 1, pp. 193–207, 2007.
[48] C. Louen, S. Ding, and C. Kandler, "A new framework for remaining useful life estimation using support vector machine classifier," in *2013 Conference on Control and Fault-Tolerant Systems (SysTol)*. IEEE, 2013, pp. 228–233.
[49] K. Aggarwal, O. Atan, A. K. Farahat, C. Zhang, K. Ristovski, and C. Gupta, "Two birds with one network: Unifying failure event prediction and time-to-failure modeling," in *2018 IEEE International Conference on Big Data (Big Data)*. IEEE, 2018, pp. 1308–1317.

BIOGRAPHIES OF AUTHORS

Alawi Alqushaibi is an academic researcher who has focused his research interests in the areas of machine learning, data science, optimization, feature selection, classification, data analytics, and image processing, specifically in Generative Adversarial Networks (GANs). He received his B.Sc. degree in Computer Networks and Security from Universiti Teknologi Malaysia in 2012, and his Master's degree by research from Universiti Teknologi PETRONAS in 2021. During his academic journey, He has acquired knowledge and skills in conducting independent research, producing academic writing, and teaching computer science courses. He can be contacted at email: alawi_18000555@utp.edu.my.

Mohd Hilmi Hasan Dr Mohd Hilmi Hasan is an associate professor and chair of the computer and information sciences department at Universiti Teknologi Petronas. He joined UTP in 2004 and his research interests in data science and data analytics led to his attachment to the Centre for Research in Data Science. His work in industry-based projects involves the whole spectrum of the data science cycle and managing big data, including the study and development of appropriate system architecture and platforms. He also carries out fundamental research in fuzzy logic type-2 for which he has secured several grants from the Malaysia government. Dr Hasan sits on the panel of assessors for the Malaysia accreditation body, and on the panel for the Malaysia Board of Technologists' Seoul Accord initiative. He can be contacted at email: mhilmi_hasan@utp.edu.my.

Said Jadid Abdulkadir is a (Senior Member, IEEE) received the B.Sc. degree in computer science from Moi University, the M.Sc. degree in computer science from Universiti Teknologi Malaysia, and the Ph.D. degree in information technology from Universiti Teknologi PETRONAS. He is currently an Associate Professor with the Department of Computer and Information Sciences, Universiti Teknologi PETRONAS. His research interests include machine learning, deep learning architectures, and applications in predictive and streaming analytics. He is serving as a Executive Committee Member for IEEE Computational Intelligence Society Malaysia chapter and the Editor-in-Chief for platform journal. He can be contacted at email: saidjadid.a@utp.edu.my.

Shakirah Mohd Taib is a lecturer and researcher at Center for Research in Data Science (CERDAS), Institute of Autonomous Systems and Autonomous Facilities (IAS), in Universiti Teknologi PETRONAS (UTP), Malaysia. She obtained a bachelor's degree in information technology from Universiti Utara Malaysia and Master of Computing from University of Tasmania, Australia. She has more than 15 years working experience at Universiti Teknologi Petronas (UTP). Her area of specialization includes data science, machine learning, knowledge discovery and information retrieval using Artificial Intelligence techniques. Shakirah is a member of international organization such as IEEE, Malaysia Board of Technologists (MBOT) and Association for Information Systems (AIS). She can be contacted at email: shakita@utp.edu.my.

Safwan Mahmood Al-Selwi received his bachelor's degree in software engineering from Taiz University, Yemen, in 2012, and his master's degree in computer applications from Bangalore University, India, in 2018. He is currently a Research Assistant in the Computer and Information Sciences department at Universiti Teknologi PETRONAS (UTP), Malaysia. His research interests include artificial intelligence, machine learning, predictive and time-series analysis, metaheuristic algorithms, and optimization. He has a total experience of 8 years both in academic institutions and in the industry. His industry working experience is related to Android applications and website development. He can be contacted at email: safwan_21002827@utp.edu.my.

Mohammed Gamal Ragab received his Bachelor of Science degree in Software Engineering from Universiti Teknologi PETRONAS in 2019, where he demonstrated a keen interest in machine learning, computer vision, and data analysis, and has a particular interest in metastatic studies. Following the completion of his undergraduate degree, Mohammed continued his studies at Universiti Teknologi PETRONAS, pursuing a master's degree by research in Machine Learning. Currently, Mohammed is continuing his academic pursuits by pursuing a Ph.D. in Information Technology at Universiti Teknologi PETRONAS. His ongoing research builds on his previous work, focusing on the development of new and innovative techniques for optimizing the performance of deep learning models. He can be contacted at email: mohd.gamal_20497@utp.edu.my.

Ebrahim Hamid Sumiea received his B.S. degree in Software Engineering from the renowned Asia Pacific University of Technology & Innovation (APU) in 2014, where he honed his skills in various programming languages and software development methodologies. He is eager to explore the fusion of technology and business, Ebrahim pursued a Masters in Management. Currently, Ebrahim is undertaking his Ph.D. at Universiti Teknologi PETRONAS, delving deeper into the realm of artificial intelligence. His research primarily revolves around Reinforcement Learning, specifically in the area of Deep Deterministic Policy Gradient (DDPG). His work aims to leverage DDPG's potential to address complex problems, demonstrating the powerful capacity of AI to transform various sectors. He can be contacted at email: ebrahim_22006040@utp.edu.my.