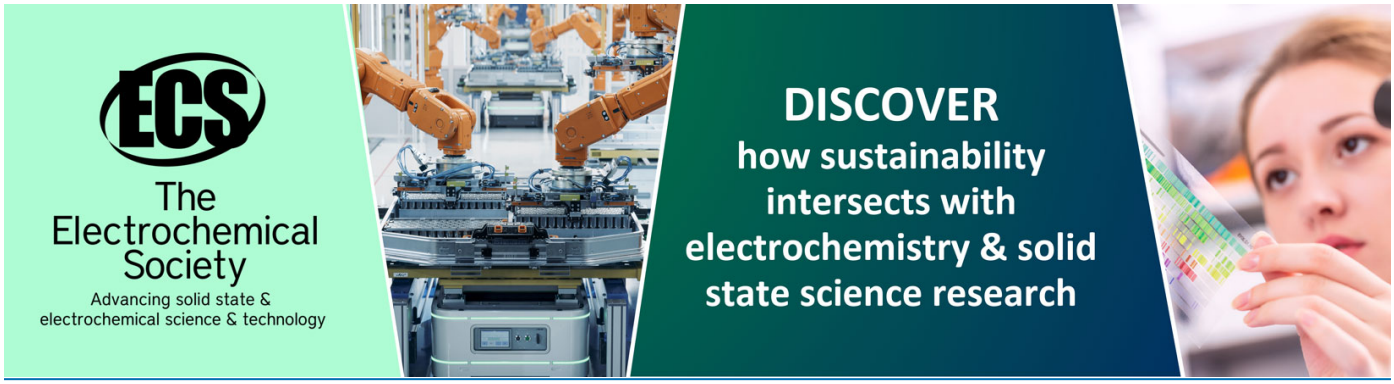
PAPER

An interpretable RUL prediction method of aircraft engines under complex operating conditions using spatio-temporal features



IOP Publishing Measurement Science and Technology
Meas. Sci. Technol. **35** (2024) 076003 (14pp) https://doi.org/10.1088/1361-6501/ad3b2c

An interpretable RUL prediction method of aircraft engines under complex operating conditions using spatio-temporal features

Jiahao Gao, Youren Wang* **and Zejin Sun**

College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, People's Republic of China

E-mail: wangyrnuaa@126.com

Received 5 February 2024, revised 19 March 2024
Accepted for publication 5 April 2024
Published 16 April 2024

Abstract

Long short-term memory (LSTM) based prediction methods have achieved remarkable achievements in remaining useful life (RUL) prediction for aircraft engines. However, their prediction performance and interpretability are unsatisfactory under complex operating conditions. For aircraft engines with high hazard levels, it is important to ensure the interpretability of the models while maintaining excellent prediction accuracy. To address these issues, an interpretable RUL prediction method of aircraft engines under complex operating conditions using spatio-temporal features (STFs), referred to as iSTLSTM, is proposed in this paper. First, we develop a feature extraction framework called Bi-ConvLSTM1D. This framework can effectively capture the spatial and temporal dependencies of sensor measurements, significantly enhancing the feature extraction capabilities of LSTM. Then, an interpretation module for STFs based on a hybrid attention mechanism is designed to quantitatively assess the contribution of STFs and output interpretable RUL predictions. The effectiveness of iSTLSTM is evidenced by extensive experiments on the C-MAPSS and N-CMAPSS datasets, confirming the superiority and reliability of our method for aircraft engine RUL prediction.

Keywords: complex operating conditions, spatial and temporal features, aircraft engines, remaining useful life prediction, attention mechanism

1. Introduction

The aircraft engine is a complex mechanical system that integrates various technologies and disciplines. Once an accident occurs, it may cause immeasurable human casualties and property damage. Furthermore, with the increasing demand for intelligent prognostic and health management (PHM) systems, it has become increasingly important to develop an accurate and reliable method for predicting remaining useful life (RUL) of aircraft engines [1].

RUL prediction is a core aspect of PHM that can be categorized into model-based, data-driven, and hybrid approaches [2]. Model-based approaches are developed by specific physical principles and failure mechanisms. Data-driven approaches aim at mining the implied relationships between sensor measurements and system deterioration through statistical, shallow learning and deep learning technologies. Hybrid approaches usually combine model-based and data-driven approaches in a complementary way for prognostic studies. Among them, model-based approaches generally have a clear physical meaning. However, due to the complexity of aircraft engine systems, it is difficult to directly establish accurate and effective physical models. Hybrid methods require extensive *a priori*

* Author to whom any correspondence should be addressed.

knowledge and balancing the effects of various models, making it challenging to construct ideal hybrid models. Recently, with the rapid development of big data and sensor technologies, the deep learning-based RUL prediction method for aircraft engines has become a research hotspot [3].

Recurrent neural networks (RNNs) are one of the primary frameworks of deep learning. Unlike feedforward neural networks, the hidden layer of RNNs simultaneously encompasses both the input layer's output and the previous output of the hidden layer. Consequently, RNNs are well-suited for handling RUL prediction tasks [4]. Nevertheless, the issue of vanishing or exploding gradients may arise in RNNs when the sequence becomes excessively long. To overcome this challenge, long short-term memory (LSTM) has emerged [5]. LSTM introduces the gating mechanism and cell structure based on RNNs, which can exclude invalid information and transmit important information over long distances. Therefore, LSTM is widely employed in the field of aircraft engine RUL prediction [6, 7]. Zheng *et al* performed comprehensive experiments on three publicly available PHM datasets, demonstrating the superior performance of LSTM compared to convolutional neural networks (CNN) and other conventional methods for RUL prediction [8]. It should be acknowledged that LSTM primarily relies on historical information and may not fully exploit subsequent contextual information. That is, LSTM can only process unidirectional input sequences, which may impose limitations on tasks that heavily rely on contextual relationships. Subsequently, a bidirectional LSTM (BLSTM) network [9] was developed to track the health state of aircraft engines and predict their RUL. The experimental results showed that BLSTM outperforms other widely employed methods including LSTM. However, it is worth noting that BLSTM faces challenges in its predictive performance under complex conditions and lacks transparency. To overcome these problems, Huang *et al* achieved accurate RUL prediction for aircraft engines under complex operating conditions by inputting a fusion of multiple sensor data and operating condition data into a BLSTM-based model [10]. After that, Chen *et al* proposed a novel multidimensional RNN based on the BLSTM framework for modeling both multidimensional monitoring data and operational condition data, which achieved RUL prediction under multiple operational conditions and fault modes [11]. Furthermore, adding an interpretability module, like the attention mechanism, has proven to be a highly effective way to improve the interpretability and accuracy of LSTM-based models in RUL prediction [12, 13]. Some researchers have demonstrated that integrating attention with LSTM improves accuracy and interpretability, while also highlighting significant degradation aspects, which is crucial for improving RUL prediction performance [14–16].

LSTM-based methods have already achieved excellent performance in aircraft engine RUL prediction, yet there are still some critical issues that impede its further deployment and acceptance in practice [17]. Specifically, (1) their feature extraction capability for nonlinear high-dimensional data is insufficient, which leads to poor prediction performance under complex operating conditions. In real cases, aircraft engines are usually operated under various operating conditions, such as different altitudes and throttle rotation angles, etc. These complex operating conditions often result in data distribution shifts, which significantly impact the predictive performance of LSTM [18]. Although previous studies have shown that integrating multiple sensor data can improve the RUL prediction performance of LSTM under complex operating conditions [10, 11], it should be noted that these methods often overlook the distribution characteristics and spatial relationships of sensors under different operating conditions. In fact, aircraft engines are typically equipped with multiple sensors that can naturally form a spatial network containing rich health status information [19]. For example, in a typical turbofan engine under normal operation, the pressure at the fan inlet is usually lower than the outlet of the high-pressure compressor. However, if the pressure at the fan inlet unexpectedly exceeds the outlet pressure of the high pressure, failure may occur. (2) Most existing methods are black-box models, which are incapable of learning interpretable features to provide a comprehensive understanding of the degradation behavior of complex systems. For high-risk equipment like aircraft engines, it is essential to ensure both high predictive accuracy and interpretability of the results [17]. Although some studies have introduced attentional mechanisms to explain the predictive results of models [12, 20], they only focus on the temporal importance while ignoring their spatial correlation. In practical engineering, the input characteristics provided by multiple sensors exhibit different contributions to the predictive results. In other words, the RUL prediction results not only have different dependencies on inputs at different time steps but also on the spatial characteristics [21].

To address the aforementioned issues, we propose an interpretable RUL prediction method for aircraft engines under complex operating conditions using spatio-temporal features (STFs), called iSTLSTM, which mainly includes two stages. The first stage involves the Bi-ConvLSTM1D framework, which effectively captures the spatial and temporal dependencies of sensor data. In the second stage, we develop a spatio-temporal feature interpretation module to quantitatively evaluate the captured STFs and output interpretable RUL predictions. Meanwhile, we employ the gird search and early stopping techniques to search for the optimal model parameters, which is a simple and intuitive way to set hyperparameters. The effectiveness of the proposed method is validated on the C-MAPSS and N-CMAPSS datasets from the NASA Prognostics Center of Excellence. The major contributions of this study are as follows:

(1) This study proposes a novel two-stage prediction method, specifically designed for Interpretable RUL predictions. A series of experiments are conducted, demonstrating that the proposed method outperforms existing state-of-the-art (SOTA) methods.
(2) A framework for extracting STFs is developed. This framework considers the positional relationship of the

sensors, which can effectively enhance the feature extraction capability of LSTM.

(3) Considering the contributions of both temporal and spatial features to the predictive results, an interpretation module based on a hybrid attention mechanism is designed that can be used to provide a better interpretation.

2. Methodology

2.1. LSTM and its variant

LSTM has achieved excellent performance in RUL prediction through the gating mechanism and memory cell structure. The key equations of LSTM can be expressed as follows:

$$\mathbf{f}_t = \sigma\left(\mathbf{W}_f \cdot \mathbf{x}_t + \mathbf{V}_f \cdot \mathbf{h}_{t-1} + \mathbf{b}_f\right) \tag{1}$$

$$\mathbf{i}_t = \sigma\left(\mathbf{W}_i \cdot \mathbf{x}_t + \mathbf{V}_i \cdot \mathbf{h}_{t-1} + \mathbf{b}_i\right) \tag{2}$$

$$\mathbf{o}_t = \sigma\left(\mathbf{W}_o \cdot \mathbf{x}_t + \mathbf{V}_o \cdot \mathbf{h}_{t-1} + \mathbf{b}_o\right) \tag{3}$$

$$\mathbf{c}_t = \mathbf{f}_t \otimes \mathbf{c}_{t-1} + \mathbf{i}_t \otimes \tanh\left(\mathbf{W}_c \cdot \mathbf{x}_t + \mathbf{V}_c \cdot \mathbf{h}_{t-1} + \mathbf{b}_c\right) \tag{4}$$

$$\mathbf{h}_t = \mathbf{o}_t \otimes \tanh\left(\mathbf{c}_t\right) \tag{5}$$

where $\mathbf{f}_t$, $\mathbf{i}_t$, and $\mathbf{o}_t$ represent the forget gate, input gate, and output gate at time step t, respectively. $\mathbf{W}_*$ and $\mathbf{V}_*$ denote the corresponding weights for the gate units and the recurrent connections. $\mathbf{b}_*$ is the bias term for the gating mechanisms. $\mathbf{h}_t$, $\mathbf{c}_t$, and $\mathbf{x}_t$ represent the hidden state, memory cell, and input data at time step t. The symbols $\otimes$ and $\bullet$ indicate the Hadamard product and matrix multiplication, respectively. σ represents the sigmoid activation function.

BLSTM networks process sequential data bidirectionally, unlike conventional LSTMs that operate unidirectionally. The network structure is illustrated in figure 1, which enhances its ability to capture global contexts by integrating both past and future information. The transition functions of BLSTM are described as follows:

$$\vec{\mathbf{h}}_t = \mathrm{LSTM}\left(\vec{\mathbf{x}}_t, \vec{\mathbf{h}}_{t-1}\right) \tag{6}$$

$$\overleftarrow{\mathbf{h}}_t = \mathrm{LSTM}(\overleftarrow{\mathbf{x}}_t, \overleftarrow{\mathbf{h}}_{t+1}) \tag{7}$$

$$\mathbf{H}_t = \vec{\mathbf{h}}_t \oplus \overleftarrow{\mathbf{h}}_t \tag{8}$$

where $\mathbf{H}_t$ represents the final output of the BLSTM cell. $\vec{\mathbf{h}}_t$ and $\overleftarrow{\mathbf{h}}_t$ denote the outputs of the forward and backward propagations of the LSTM, respectively. The concatenation operation is denoted by $\oplus$.

2.2. Proposed method

The flowchart of iSTLSTM is illustrated in figure 2, which mainly consists of two parts: spatio-temporal feature extraction framework and spatio-temporal feature interpretation module. First, the input data under different working conditions are preprocessed to provide suitable inputs for the model. Then, Bi-ConvLSTM1D is utilized to simultaneously extract the temporal and spatial features of the data. Finally, an interpretable model based on a hybrid attention mechanism is used to quantitatively evaluate the extracted STFs and output interpretable RUL prediction results. The following sections provide a detailed description of these parts.

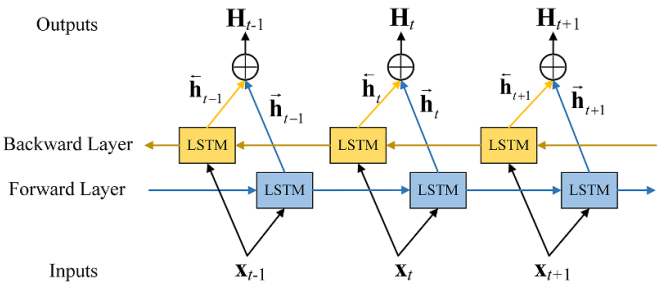


Figure 1. BLSTM network structure.

2.2.1. STF extraction framework.

To preserve the spatial integrity of degradation features in the data, we developed the Bi-ConvLSTM1D framework. This framework embeds 1D convolutional operations into the gating unit of LSTM instead of the conventional matrix-based multiplication method. Additionally, to improve the ability to mine the degradation trend of aircraft engines, we extract STFs from both directions of the sequence. Unlike the BLSTM model, we utilize the ConvLSTM1D module as the basic unit in our work, as depicted in figure 3. The working principle of ConvLSTM1D can be described as follows:

$$\mathbf{f}_t = \sigma\left(\mathbf{W}_f * \mathbf{x}_t + \mathbf{V}_f * \mathbf{h}_{t-1} + \mathbf{b}_f\right) \tag{9}$$

$$\mathbf{i}_t = \sigma\left(\mathbf{W}_i * \mathbf{x}_t + \mathbf{V}_i * \mathbf{h}_{t-1} + \mathbf{b}_i\right) \tag{10}$$

$$\mathbf{o}_t = \sigma\left(\mathbf{W}_o * \mathbf{x}_t + \mathbf{V}_o * \mathbf{h}_{t-1} + \mathbf{b}_o\right) \tag{11}$$

$$\mathbf{c}_t = \mathbf{f}_t \otimes \mathbf{c}_{t-1} + \mathbf{i}_t \otimes \tanh\left(\mathbf{W}_c * \mathbf{x}_t + \mathbf{V}_c * \mathbf{h}_{t-1} + \mathbf{b}_c\right) \tag{12}$$

$$\mathbf{h}_t = \mathbf{o}_t \otimes \tanh\left(\mathbf{c}_t\right) \tag{13}$$

where $\otimes$ and $*$ denote Hadamard product and convolution operations, respectively. $\mathbf{f}_t$, $\mathbf{i}_t$, and $\mathbf{o}_t$ indicate the corresponding various gating mechanisms of the ConvLSTM1D unit. $\mathbf{W}_* \in \mathbb{R}^{N \times M}$, $\mathbf{V}_* \in \mathbb{R}^{N \times N}$ and $\mathbf{b}_* \in \mathbb{R}^{N}$ are the shared weights and bias vectors of the model. M and N are the number of sensors and hidden state units, respectively. $\mathbf{x}_t \in \mathbb{R}^{M}$ denotes the input vector. $\mathbf{h}_t \in \mathbb{R}^{M \times N}$ is the hidden states output of the ConvLSTM1D unit.

The main distinction between ConvLSTM1D and LSTM is the operation of the gating unit weight matrices with the inputs $\mathbf{x}_t$ and the previous hidden state $\mathbf{h}_{t-1}$. While LSTM uses a Hadamard operation that directly shuffles the input data, potentially disrupting its spatial characteristics, ConvLSTM1D employs convolutional operations instead (refer to equations (9)–(12)). Thus, our method can effectively capture the temporal and spatial dependencies of the sensor measurements.

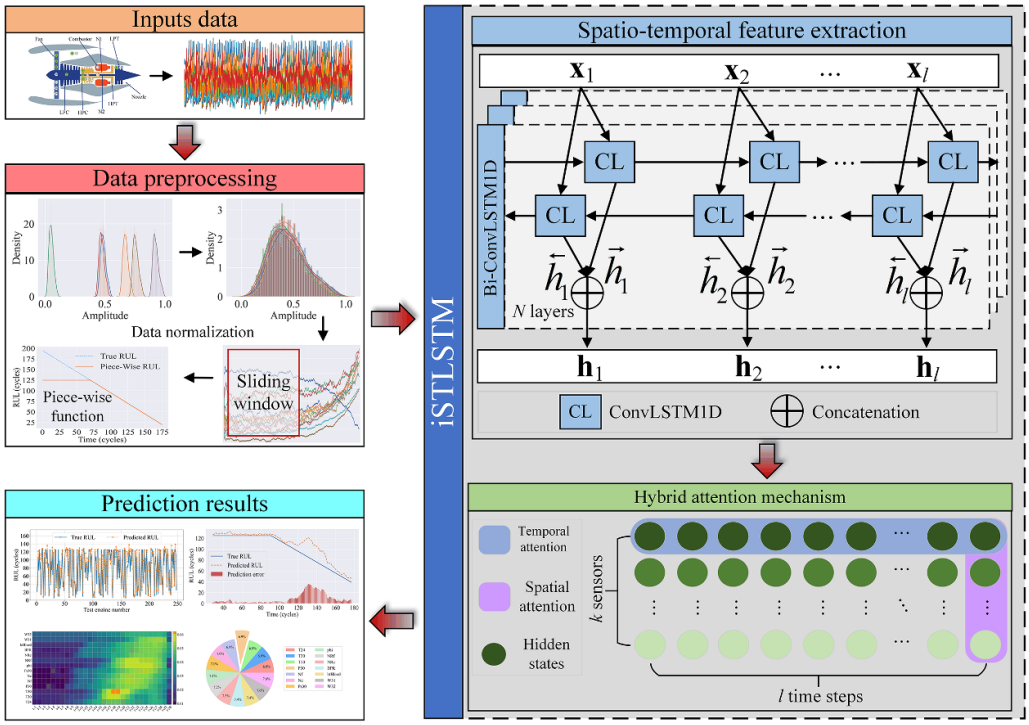


Figure 2. Flowchart of the proposed method.

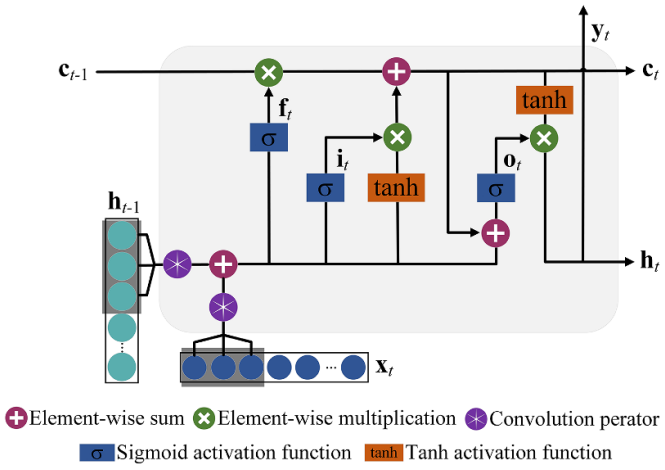


Figure 3. Structure of ConvLSTM1D.

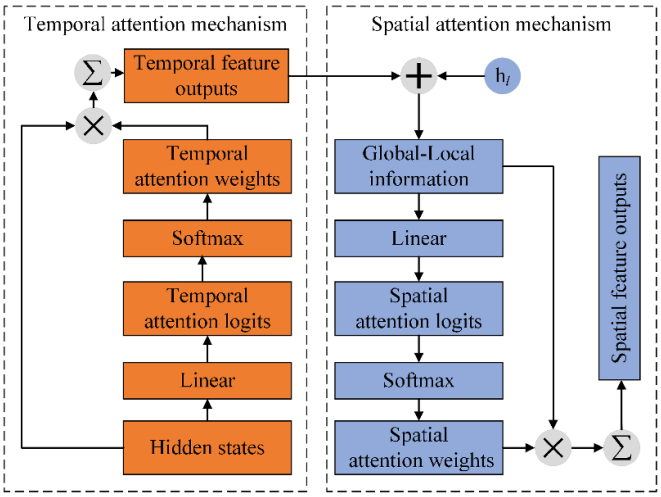


Figure 4. Schema of the hybrid attention module.

2.2.2. STF interpretation module. To enhance the interpretability of the model, an interpretation module based on a hybrid attention mechanism is designed in this study. This interpretation module considers the correlation of data in both temporal and spatial dimensions as illustrated in figure 4. Temporal attention is first employed to capture the contributions of each sensor at every time step by analyzing the hidden state sequences associated with the monitoring data. Spatial attention is then calculated to ascertain the contribution of each sensor to the final prediction outcomes.

Assuming that the STFs extracted by Bi-ConvLSTM1D are represented as hidden state $\mathbf{H} = \{\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_l\}$ at different time steps, where the sequence of hidden states about sensor m is denoted as $\{h_1^m, h_2^m, \ldots, h_l^m\}$. Based on the temporal attention mechanism, the score $\mathbf{T}_{si}$ for the feature at the i_{th} time step can be expressed as:

$$\mathbf{T}_{si} = \tanh\left(\mathbf{W}_T \cdot \mathbf{h}_i + \mathbf{b}_T\right) \tag{14}$$

where $\mathbf{W}_T$ and $\mathbf{b}_T$ are the weight matrix and bias vector of the temporal attention mechanism, respectively. $\mathbf{h}_i$ denotes the hidden state at the ith time step. After calculating the score

of $\mathbf{h}_i$, it is normalized by the softmax function to obtain the weight $\mathbf{T}_{ai}$ at this moment. The expression is given as follows:

$$\mathbf{T}_{ai} = \text{softmax}\left(\mathbf{T}_{si}\right) = \frac{\exp\left(\mathbf{T}_{si}\right)}{\sum_i \exp\left(\mathbf{T}_{si}\right)}. \tag{15}$$

Then, the output $\mathbf{O}_T$ of the temporal attention mechanism is obtained by weighting the matrix $\mathbf{H}$ with the weights $\mathbf{Ta}$ for different time steps, where $\mathbf{Ta} = \{\mathbf{T}_{a1}, \mathbf{T}_{a2}, \ldots, \mathbf{T}_{al}\}$. The expression is defined as follows:

$$\mathbf{O}_T = \mathbf{H} \otimes \mathbf{Ta} \tag{16}$$

where $\otimes$ is the Hadamard product indicating element-wise multiplication. To better capture long-term dependencies in the sequence data, we fuse the output $\mathbf{O}_T$ with the hidden state $\mathbf{h}_l$ of the last time step to obtain $\mathbf{G}_L$ that encompasses both global and local information. $\mathbf{G}_L$ is then used as the input for the spatial attention mechanism, providing richer contextual information to comprehensively understand the semantic and structural relationships in the input data. More specifically, the spatial attention mechanism is computed using the following formula:

$$\mathbf{G}_L = \mathbf{O}_T \oplus \mathbf{h}_l \tag{17}$$

$$\mathbf{S}_{sm} = \tanh\left(\mathbf{W}_S \cdot \mathbf{G}_L + \mathbf{b}_S\right) \tag{18}$$

$$\mathbf{S}_{am} = \text{softmax}\left(\mathbf{S}_{sm}\right) = \frac{\exp\left(\mathbf{S}_{sm}\right)}{\sum_i \exp\left(\mathbf{S}_{sm}\right)} \tag{19}$$

$$\mathbf{O} = \mathbf{G}_L \otimes \mathbf{Sa} \tag{20}$$

where $\otimes$ and $\oplus$ denote the Hadamard product and concatenation operation, respectively. $\mathbf{W}_S$ and $\mathbf{b}_S$ represent the weight matrix and bias vector of the spatial attention mechanism. $\mathbf{S}_{sm}$ is the score of the m_{th} sensor monitoring data. $\mathbf{S}_{am}$ denotes the importance weight of the spatial feature for the m_{th} sensor. $\mathbf{Sa} = \{\mathbf{S}_{a1}, \mathbf{S}_{a2}, \ldots, \mathbf{S}_{aM}\}$ represents the spatial attention feature maps for different sensors. $\mathbf{O}$ denotes the output of the final hybrid attention mechanism.

3. Experiments study

All experiments are conducted on a PC with an Intel Xeon Silver 4210 R 2.40 GHz CPU and an NVIDIA GeForce RTX3090 GPU to validate the effectiveness of iSTLSTM.

3.1. Data description

3.1.1. CMAPSS dataset. The C-MAPSS dataset [22] consists of four subsets as presented in table 1. Each subset includes a training set, test set, and actual RUL values. The training set comprises run-to-failure data, while the testing set contains partial pre-failure data. Each group of data includes three operational settings and monitoring data from 21 sensors, as described in table 2. Figure 5 shows the sensor installation locations and the major components of the aircraft engine, which include the Combustor, Nozzle, and five rotating parts: the fan, low-pressure turbine (LPT), low-pressure compressor (LPC), high-pressure compressor (HPC), and high-pressure turbine (HPT).

Since there are some sensors with irregular measurements that cannot offer useful deterioration information, we selected 14 sensor data to train the model based on signal correlation, monotonicity, and signal-to-noise ratio. These sensors are T24, T30, T50, P30, Nf, Nc, Ps30, phi, NRf, NRc, BPR, htBleed, W31 and W32, respectively [18, 23].

3.1.2. N-CMAPSS dataset. The N-CMAPSS [24] is the PW4090 turbofan engine run-to-failure dataset under real flight conditions, which covers various flight states such as climb, cruise, and descent. Here, we employ the widely used DS02 subset from the N-CMAPSS dataset to evaluate the proposed method. This dataset contains a total of nine units and each unit has a different initial degradation state. Table 3 summarizes the number of samples m_i, transition time t_s, end-of-life time t_{EOL}, flight class Fc, and the failure modes for each unit in DS02. For more details about the dataset please refer to [24].

It is worth noting that the N-CMAPSS dataset is recorded at a 1 Hz sampling rate, which results in a large dataset. To improve the iteration speed of training and to obtain a larger receptive field, the data sampling rate is set to 0.01 Hz in this paper, so the total number of training and testing samples of the model is 0.053 and 0.012 M, respectively. In the selection of input variables, some health parameters have constant values that are irrelevant to the health state of the engines and cannot characterize their degradation process. Therefore, a total of 31 input variables are selected for RUL prediction in this case, including physical measurement signals, virtual sensor measurements, and three health parameters related to the engines.

3.2. Data preprocessing

To mitigate the influence of multiple operating conditions, we employ the conditional normalization technique, which involves computing the mean and variance of the monitoring data under different operating conditions. Subsequently, the data is normalized to a unified scale based on the corresponding operating condition. The computation process is described as follows [25]:

$$\mathbf{x}_c = \frac{\left(\mathbf{s}_c - \mu_c\right)}{\sigma_c} \tag{21}$$

where c represents the operating condition. $\mathbf{s}_c$ denotes the sensor measurements at condition c. μ_c and σ_c are the mean and variance of the monitoring data at the corresponding operating condition. $\mathbf{x}_c$ is the normalized data. In the N-CMAPSS dataset, $\mathbf{x}_c$ represents the entire dataset due to the complexity and variability of operating conditions.

Table 1. Description of the C-MAPSS dataset.

Dataset	Operating conditions	Fault types	Number of engines in the training set	Number of engines in the test set
FD001	1	1	100	100
FD002	6	1	260	259
FD003	1	2	100	100
FD004	6	2	249	248

Table 2. Description of C-MAPSS output sensor measurements.

Index of sensor	Symbol	Description	Units
1	T2	Total temperature at fan inlet	°R
2	T24	Total temperature at LPC outlet	°R
3	T30	Total temperature at HPC outlet	°R
4	T50	Total temperature at LPT outlet	°R
5	P2	Pressure at fan inlet	psia
6	P15	Total pressure in bypass-duct	psia
7	P30	Total pressure at HPC outlet	psia
8	Nf	Physical fan speed	rpm
9	Nc	Physical core speed	rpm
10	erp	Engine pressure ratio (P50/P2)	—
11	Ps30	Static pressure at HPC outlet	psia
12	phi	Ratio of fuel flow to Ps30	pps/psi
13	NRf	Corrected fan speed	rpm
14	NRc	Corrected core speed	rpm
15	BPR	Bypass ratio	—
16	farB	Burner fuel–air ratio	—
17	htBleed	Bleed enthalpy	—
18	Nf_dmd	Demanded fan speed	rpm
19	PCNfR_dmd	Demanded corrected fan speed	rpm
20	W31	HPT coolant bleed	lbm/s
21	W32	LPT coolant bleed	lbm/s

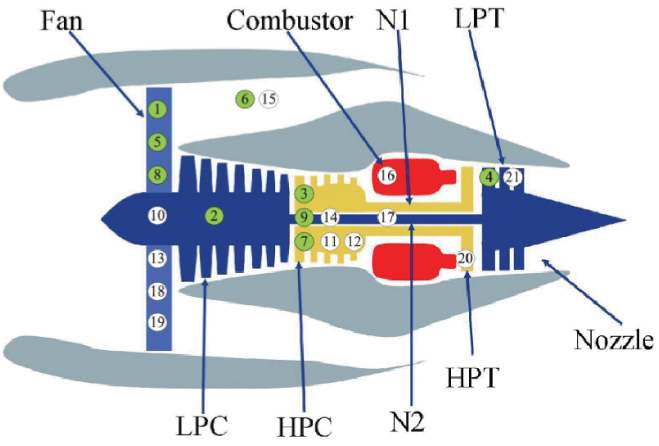


Figure 5. Simplified diagram of the engine simulated in the C-MAPSS dataset.

Subsequently, we employ a sliding window technique [26, 27] to segment the data so that each sample consists of a continuous sequence of measurements, which preserves the correlation between sensor measurements. In the C-MAPSS dataset, we set the sliding window size and sliding step to 30 and 1. In the N-CMAPSS dataset, the sliding window size and sliding step are 50 and 1, respectively.

Finally, considering the long-tailed distribution of engine degradation trajectories, we apply the piece-wise linear function to create sample labels [8, 28]. Specifically, if the actual RUL exceeds the pre-defined threshold, the maximum RUL is automatically assigned. To facilitate fair comparisons with other models, we set the maximum RUL to 125 cycles in this study [26, 29, 30].

Table 3. Description of the N-CMAPSS dataset.

Dataset	Unit	m_i	t_s	t_{EOL}	Fc	Failure mode
Training set	2	0.85 M	17	75	3	HPT
	5	1.03 M	17	89	3	HPT
	10	0.95 M	17	82	3	HPT
	16	0.77 M	16	63	3	HPT + LPT
	18	0.89 M	17	71	3	HPT + LPT
	20	0.77 M	17	66	3	HPT + LPT
Test set	11	0.66 M	19	59	3	HPT + LPT
	14	0.16 M	36	76	1	HPT + LPT
	15	0.43 M	24	67	2	HPT + LPT

3.3. Evaluation metrics

In order to quantitatively assess the performance of the model, the root mean square error (RMSE) and Score are used in this study. These two metrics are defined as follows:

$$\mathrm{RMSE} = \sqrt{\frac{1}{p}\sum_{i=1}^{p}\left(\hat{y}_i - y_i\right)^2} \tag{22}$$

$$\mathrm{Score} = \begin{cases} \sum_{i=1}^{p} e^{-\left(\frac{\hat{y}_i - y_i}{13}\right)} - 1, & \hat{y}_i - y_i < 0 \\ \sum_{i=1}^{p} e^{\left(\frac{\hat{y}_i - y_i}{10}\right)} - 1, & \hat{y}_i - y_i \geqslant 0 \end{cases} \tag{23}$$

where $\hat{y}_i$ and y_i represent the predicted and true RUL values of the ith sample, respectively. p denotes the number of samples. Equations (22) and (23) show that both metrics evaluate the deviation between the predicted and true RUL, with lower values indicating smaller prediction errors. Among them, RMSE treats all predictions equally, whereas Score assigns a greater penalty to late RUL predictions compared to early predictions.

3.4. Hyperparameters setting

The performance of the model is closely related to the hyperparameter configuration. To obtain the optimal values we employ a grid search technique [31] and early stopping technique to determine the hyperparameters. To mitigate the risk of overfitting, we divide 20% of the training data as a validation set and monitor the model optimization process using the loss of the validation set. If the validation loss fails to decrease after 20 epochs, the model stops training. The remaining parameters are determined by the grid search technique. Table 4 lists the specific settings of the hyperparameters for the proposed method. Despite the search space not being fully comprehensive, such a search strategy allows for reasonable and general conclusions to be drawn about the predictive performance of each dataset, Similar strategies can also be found in [32, 33].

Table 4. List of hyperparameters for the proposed method.

Parameters	C-MAPSS	N-CMAPSS
Batch size	256	512
Learning rate	0.001	0.001
Optimizer	Adam	Adam
Loss function	RMSE	RMSE
Convolution kernel sizes	[5, 7, 9]	[9, 15, 21]
Hidden layer units	[32, 64, 128]	[32, 64, 128]
Bi-ConvLSTM1D layers	[1, 2, 3]	[1, 2]

3.5. Experimental results of the C-MAPSS dataset

3.5.1. Comprehensive performance validation. Figure 6 illustrates the prediction results of iSTLSTM on the C-MAPSS datasets for a fleet of engines. We can see that the predicted RUL is largely consistent with the true RUL values, especially for the FD001 and FD003 datasets. Moreover, our method shows superior prediction performance on the FD002 dataset compared to the FD004 dataset. The discrepancy may arise from the fact that FD001 and FD003 are under a single operating condition, making their deterioration trend more predictable. Meanwhile, the FD002 dataset has fewer failure modes and its degradation trend is simpler and more pronounced compared to FD004. Further observation reveals that the predicted RUL values for individual engines show slight deviations in figure 6(d). One possible explanation for this phenomenon is that there are both HPC and Fan failure modes in the FD004 dataset. Specifically, the engine thrust will decrease significantly when the fan fails. Therefore, its degradation tendency is relatively clearer. As for HPC faults, their failure modes depend on various factors (e.g. operating conditions, ambient temperature and humidity, etc). Therefore, its degradation trend is more complex and unstable.

Additionally, we present the prediction error distribution plots of the proposed method on C-MAPSS datasets in figure 7. It can be seen that the prediction errors of iSTLSTM are mainly centered around zero on both datasets, indicating the effectiveness of our method in accurately predicting the RUL of aircraft engines under complex operating conditions. Although there are prediction biases for individual engines in the FD004 dataset, their predictions are smaller than the true RUL values. From the perspective of equipment safety, this result aligns with the requirements of high-risk equipment like aircraft engines.

3.5.2. Individual performance analysis. To further validate the predictive performance of the proposed method for the complete degradation processes of the individual engine, we

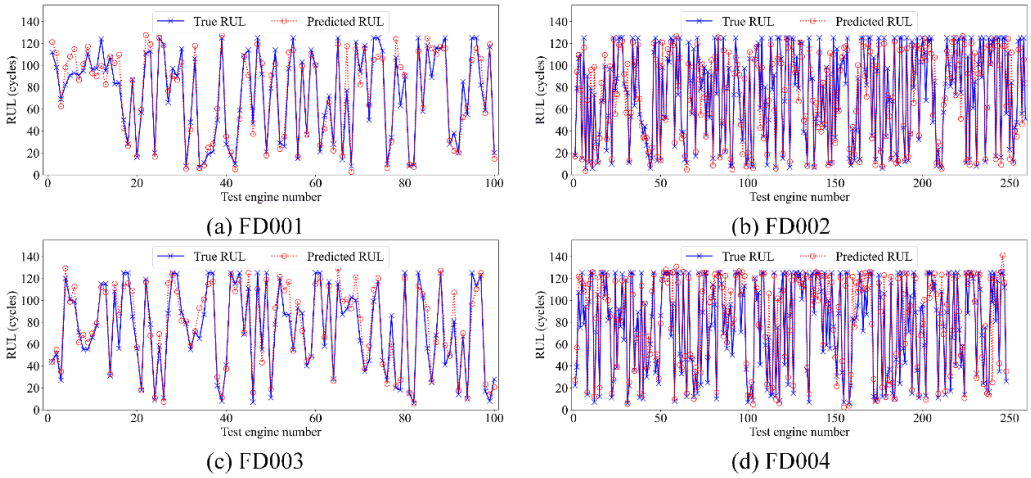


Figure 6. Predicted results on a fleet of engines.

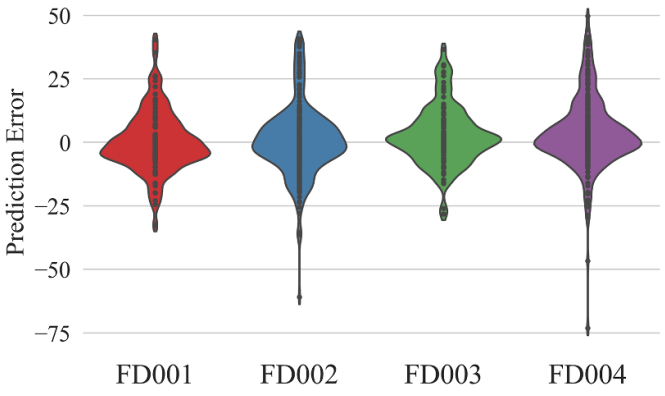


Figure 7. Prediction error distribution of the proposed method.

randomly selected four engines from the FD002 and FD004 datasets to analyze the effectiveness of iSTLSTM, as shown in figures 8 and 9. It can be observed that the predicted and true RUL values for each engine in both datasets exhibit a basic consistent degradation trend, demonstrating iSTLSTM can effectively predict the complete degradation processes of the individual engine. Furthermore, we observe that the prediction errors are small in the initial stage, while the errors increase significantly after the engine starts to degrade. This phenomenon can be attributed to the fact that the engine is predominantly in a healthy state during the initial phase with more immediate and predictable characteristics, as shown in figures 8(c) and 9(b). In the mid-stage of engine degradation, there are limited degradation features for the model to learn, so the prediction error increases. As the engine approaches failure, the degradation features become more evident and abundant, thus the prediction error is reduced. Given the higher significance of the later stages of the lifecycle for health management in practical applications [23], our method remains of considerable engineering value in this study.

3.6. Experimental results of the N-CMAPSS dataset

Figure 10 shows the prediction results of the proposed method on the N-CMAPSS dataset. In figure 10, the shaded area represents the uncertainty bounds of the RUL prediction in each cycle and the red horizontal dashed line corresponds to the prediction error of ± 5 cycles. As illustrated in figure 10(a), the RUL prediction curves fluctuate widely during the early stages. This may be because the prediction uncertainty is larger when the engine stays in the early stage of degradation, which is demonstrated in figure 10(b). However, the predicted RUL for each test unit is approaching the true RUL as the flight time increases. As shown in figure 10(b), the prediction errors for all test units converge toward zero as the engine failure time approaches. In summary, the predicted results of each engine unit are similar to the degradation trajectory of the true RUL, which demonstrates the effectiveness of the proposed method. Specifically, the RMSE and Score values of different test units are listed in table 5. It is worth noting that the units used for training on the DS02 dataset are entirely of long-distance flights ($Fc = 3$), whereas the flight profiles of Unit 14 are significantly different from those of the training units, which has the shorter-operated flights ($Fc = 1$). Therefore, it would be relatively more difficult to make accurate predictions for Unit 14.

4. Model analysis

4.1. Interpretability analysis

In order to effectively explain the prediction results of the model, we developed a hybrid attention mechanism. The average weight information of STFs learned by the iSTLSTM on the two datasets is shown in figure 11. We can see that the contribution of temporal and spatial features is not fixed but dynamically evolving. As depicted in figure 11, our method pays more attention to the later time steps. This phenomenon

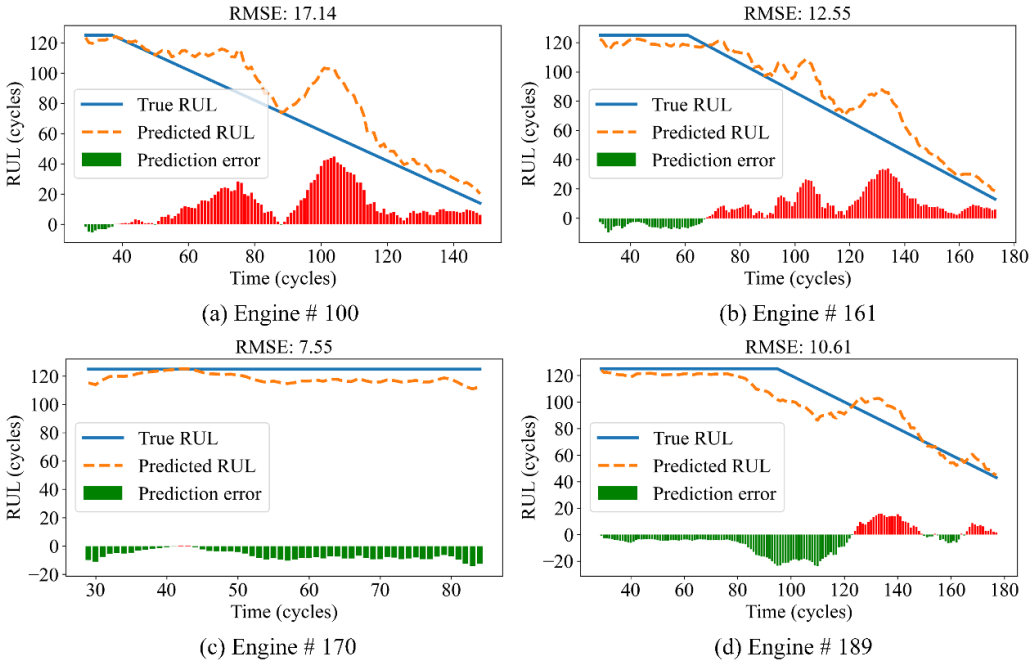


Figure 8. Predicted results for individual engines on the FD002 dataset.

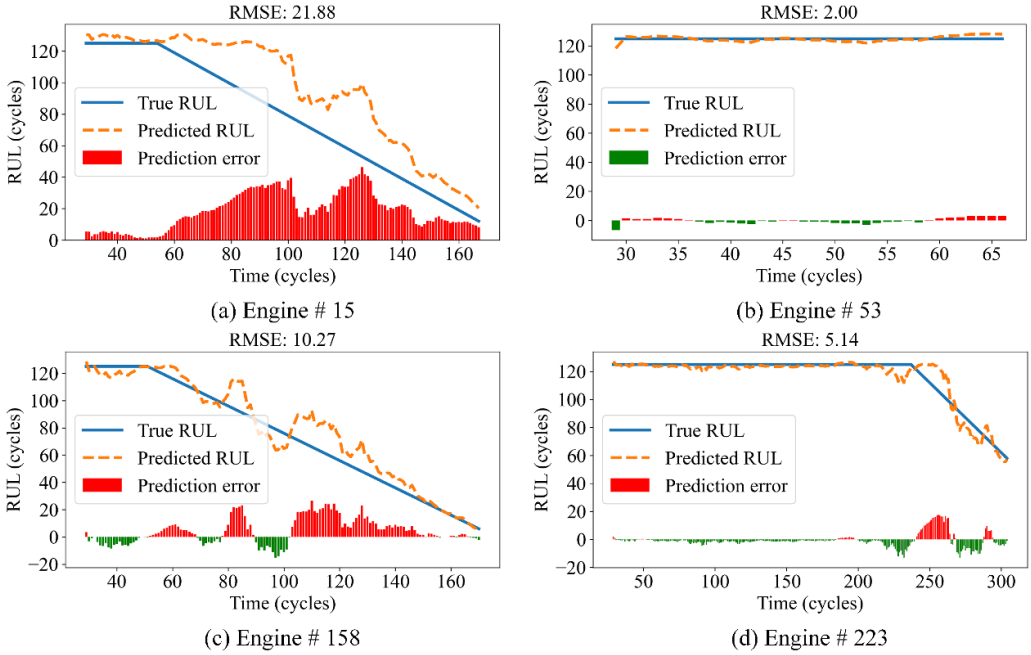


Figure 9. Predicted results for individual engines on the FD004 dataset.

is consistent with our common sense that neural network models tend to give higher importance to feature changes between adjacent time steps. In addition, we can still infer that they will contribute differently to the final RUL prediction, which is achieved by using our hybrid attention mechanism module.

Figure 12 illustrates the average weight information of the spatial features learned by our method, providing insights into the contributions of sensor measurements at various locations to the final results. From figure 12(a), it can be noticed that the P30 installed at the HPC location has the highest weighting of sensor measurements due to the failure mode of the FD002 dataset being an HPC failure. Figure 12(b) shows that the Nc, Nf, and Ps30 used to measure physical core speed, physical fan speed, and static pressure at the HPC outlet contribute significantly to the predicted results. This is because the failure modes

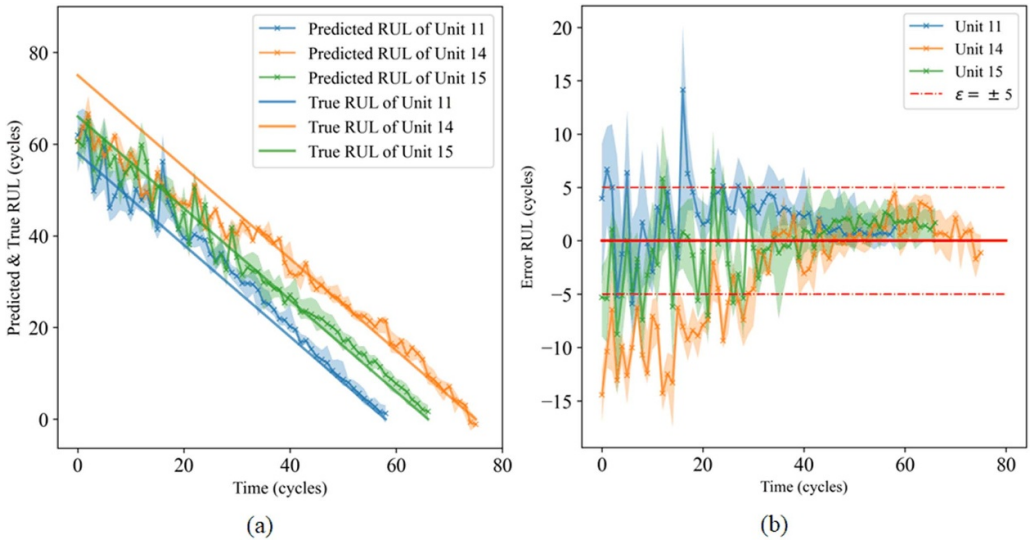


Figure 10. Predicted results for each test unit on the DS02 dataset.

Table 5. Performance of different units.

Metric	Unit 11	Unit 14	Unit 15	All units
RMSE	3.88	6.05	3.55	4.11
Score	2502.24	784.68	1251.40	4538.32

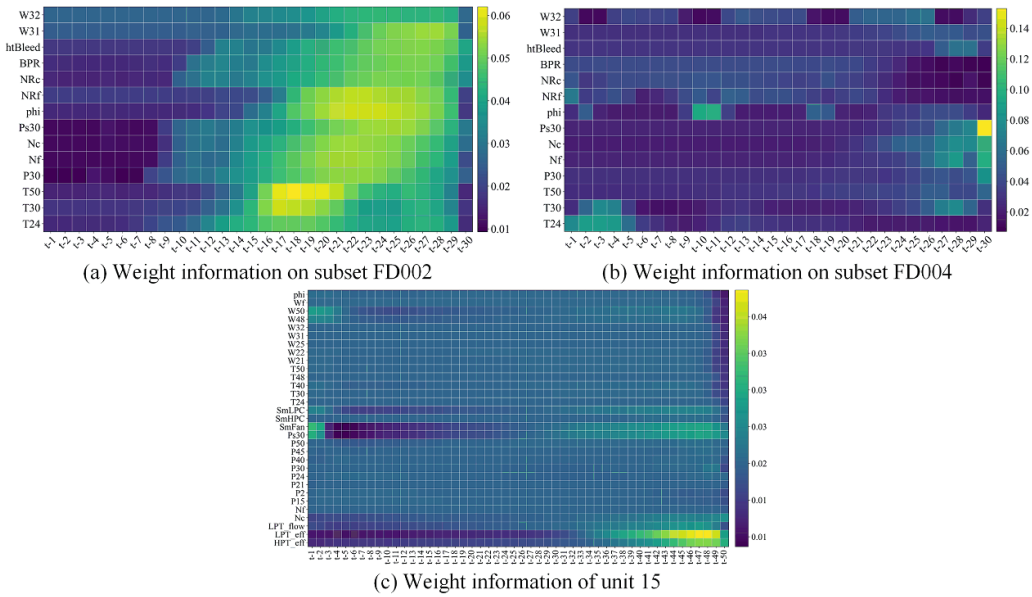
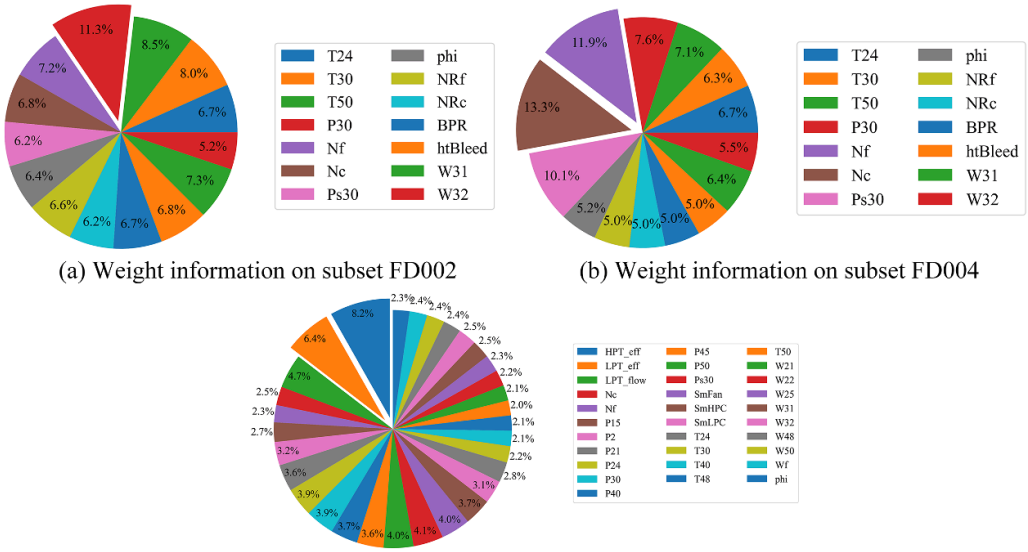


Figure 11. Average weight of spatio-temporal features.

of the FD004 dataset are associated with HPC and fan failure. As shown in figure 12(c), our method mainly focuses on the HPT efficiency modifier (HPT_eff) and LPT efficiency modifier (LPT_eff) parameters, which happens because the failure modes of unit 15 are HPT and LPT failures. The above analysis demonstrates that iSTLSTM has good interpretability and provides a clear understanding of the various factors that influence its predictions.

4.2. Impact of the hyperparameters

The impact of convolutional kernel size, hidden layer units, and the number of Bi-ConvLSTM1D layers on the N-CMAPSS dataset are analyzed in figure 13, respectively. It can be seen that the RMSE of the N-CMAPSS dataset is smallest when the hidden layer units, convolutional kernel size, and the number of Bi-ConvLSTM1D layers are 32, 15, and 2,

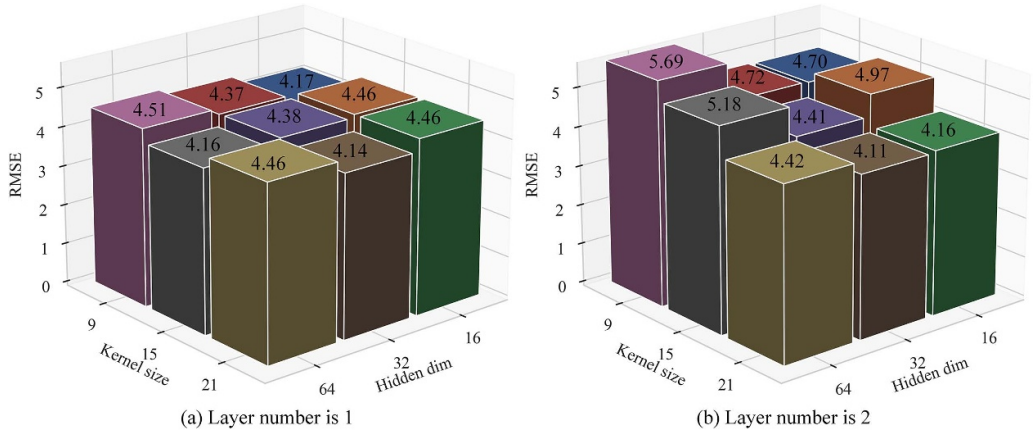


(a) Weight information on subset FD002

(b) Weight information on subset FD004

(c) Weight information of unit 15

Figure 12. Average weight of spatial features.



(a) Layer number is 1

(b) Layer number is 2

Figure 13. Experimental results with different hyperparameters.

respectively, with other hyperparameters unchanged. Further analysis reveals a decrease in RMSE as the number of network layers increases for other combination styles. This trend could be attributed to the exponential increase in model parameters and complexity with a higher number of layers, potentially leading to model overfitting. Meanwhile, we can see that the RMSE value is generally smaller with larger convolutional kernel size when the number of network layers is 2. This finding would suggest that the larger the size of the convolution kernel, the larger the sensory field of the model, thus facilitating the extraction of spatial features from the data.

4.3. Impact of the sliding window size

To explore the impact of sliding window size, we implement the proposed method with different window sizes on the C-MAPSS dataset. As shown in figure 14, we can see that the RMSE and score of iSTLSTM on both datasets initially decrease as the window size increases. However, these two metrics increase when the window size exceeds 30, and performance degrades on the FD002 and FD004 datasets. This may be because the small sliding window contains limited degradation information, resulting in poor performance. When the sliding window is too large, the sample data becomes relatively more complex, which may lead to the model suffering from underfitting. Therefore, we set the sliding window size to 30 in this paper for the C-MAPSS dataset.

4.4. Comparisons with SOTA methods

On the C-MAPSS dataset, we compare with the advanced LSTM-based methods [10, 26, 34–38] published over the past few years, along with the current SOTA methods [29, 30, 39–42]. The comparison results between our proposed method and these methods in terms of RMSE and Score metrics are listed in tables 6 and 7. It can be seen that the proposed method

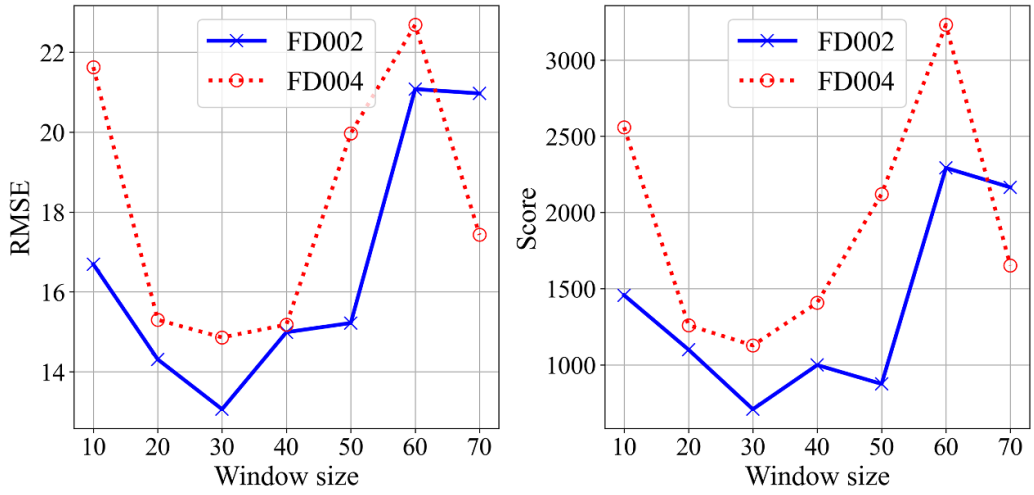


Figure 14. Experimental results under different window sizes.

Table 6. The RMSE of different methods on the C-MAPSS dataset.

Methods		FD001	FD002	FD003	FD004	Average
Advanced LSTM-based methods	BLSTM [10]	N/A	25.11	N/A	26.61	25.86
	Bi-level LSTM [26]	11.80	23.14	12.37	23.38	17.67
	HDNN [34]	13.017	15.24	12.22	18.156	14.66
	LSTM-MLSA [35]	11.567	14.02	12.134	17.21	13.73
	AEQRNN [36]	N/A	19.10	N/A	20.67	19.89
	SCAT-LSTM [37]	12.10	16.90	12.14	21.93	15.77
	BiGRU-TSAM [38]	12.56	18.94	12.45	20.47	16.11
Current SOTA methods	CA-Transformer [29]	12.25	17.08	13.39	19.86	15.65
	RVE [30]	13.42	14.92	12.51	16.37	14.31
	DAST [39]	11.43	15.25	11.32	18.36	14.09
	MTSTAN [40]	**10.97**	16.81	**10.90**	18.85	14.38
	KGHM [41]	13.18	13.25	13.54	19.96	14.98
	MSIDSN [42]	11.74	18.26	12.04	22.48	16.13
Proposed	iSTLSTM	11.92	**13.06**	11.85	**14.86**	**12.92**

Table 7. The Score of different methods on the C-MAPSS dataset.

Methods		FD001	FD002	FD003	FD004	Average
Advanced LSTM-based methods	BLSTM [10]	N/A	4793	N/A	4971	4882
	Bi-level LSTM [26]	194	3771	224	3492	1920.25
	HDNN [34]	245	1282.42	287.72	1527.42	835.64
	LSTM-MLSA [35]	252.86	899.18	370.39	1558.48	770.23
	AEQRNN [36]	N/A	3220	N/A	4597	3908.5
	SCAT-LSTM [37]	207	1267	248	3310	1258
	BiGRU-TSAM [38]	213.35	2264.13	232.86	3610.34	1580.17
Current SOTA methods	CA-Transformer [29]	198	1575	290	1741	951
	RVE [30]	323.82	1379.17	256.36	1845.99	951.34
	DAST [39]	203.15	924.96	**154.92**	1490.72	693.44
	MTSTAN [40]	**175.36**	1154.36	188.22	1446.29	741.06
	KGHM [41]	250.99	1131.03	333.44	3356.10	1267.89
	MSIDSN [42]	205.55	2046.55	196.42	2910.73	1339.81
Proposed	iSTLSTM	251.87	**708.79**	247.41	**1127.59**	**583.92**

Table 8. Comparison with different methods on the N-CMAPSS dataset.

Methods	RMSE
DGPs [43]	7.31
CNN [27]	4.14
SCTA-LSTM [37]	4.15
DLformer [44]	6.79
ALSTMP [45]	5.10
iSTLSTM (Proposed)	**4.11**

achieves the best performance in both metrics with complex operating conditions. Compared to the advanced LSTM-based approaches, our method achieves a minimum reduction of 6.85% and 13.65% in the RMSE metric on the FD002 and FD004 datasets, respectively. In addition, our method still reduces at least 1.43% and 9.22% compared to the current SOTA methods. Similarly, in terms of the Score metric, the prediction error of our method remains significantly smaller compared to both advanced LSTM-based methods and current SOTA methods. Although the results of iSTLSTM are not optimal under a single operating condition, our method significantly outperforms other methods regardless of the average metric reflecting the comprehensive performance of the model or the typical operating conditions of the aircraft engines. On the N-CMAPSS dataset, we also compare with the related SOTA methods [27, 37, 43–45] published in recent years. The results are presented in table 8. It can be seen that iSTLSTM has the smallest RMSE value, indicating that our method still outperforms all other methods.

5. Conclusion

In this study, we propose an interpretable RUL prediction method for aircraft engines based on STFs, effectively enhancing LSTM's feature extraction and achieving interpretable predictions. The experimental results on NASA's C-MAPSS and N-CMAPSS datasets demonstrate that our method can effectively predict the RUL of aircraft engines and their complete degradation process and provide a clear understanding of the various influencing factors. Moreover, compared with related SOTA methods, the proposed method achieves satisfactory results, especially under complex working conditions. Therefore, our method can be better applied to aircraft engine RUL prediction.

Although our method exhibits some interpretability in aircraft engine RUL prediction, it remains only a preliminary step. There exists a considerable gap before achieving practical industrial application, especially in the context of multiple failure modes. Hence, in future research, we will focus on the degradation mechanisms of aircraft engines. Based on its physical principles, a physical information neural network model is designed so that it can learn data characteristics while also following these basic physical laws. In this way, the model can not only improve the accuracy of prediction but also enhance the interpretability of the model.

Data availability statements

The data that support the findings of this study are openly available at the following URL/DOI: https://doi.org/10.1109/PHM.2008.4711414.

Acknowledgments

This study was supported by the Aviation Science Foundation (No. 20183352031).

Conflict of interest

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.

Credit author statement each

Jiahao Gao: Conceptualization, Methodology, Writing—original draft, Writing—review & editing. **Youren Wang:** Conceptualization, Supervision, Writing—review & editing. **Zejin Sun:** Conceptualization, Writing—review & editing.

ORCID iDs

Youren Wang https://orcid.org/0000-0001-9796-2342
Zejin Sun https://orcid.org/0000-0001-6954-5568

References

[1] Javed K, Gouriveau R and Zerhouni N 2017 State of the art and taxonomy of prognostics approaches, trends of prognostics applications and open issues towards maturity at different technology readiness levels *Mech. Syst. Signal Process.* **94** 214–36
[2] Lee J, Wu F, Zhao W, Ghaffari M, Liao L and Siegel D 2014 Prognostics and health management design for rotary machinery systems—reviews, methodology and applications *Mech. Syst. Signal Process.* **42** 314–34
[3] Che C, Wang H, Fu Q and Ni X 2019 Combining multiple deep learning algorithms for prognostic and health management of aircraft *Aerosp. Sci. Technol.* **94** 105423
[4] Chen J, Jing H, Chang Y and Liu Q 2019 Gated recurrent unit based recurrent neural network for remaining useful life prediction of nonlinear deterioration process *Reliab. Eng. Syst. Saf.* **185** 372–82
[5] Hochreiter S and Schmidhuber J 1997 Long short term memory *Neural Comput.* **9** 1735–80
[6] Sherstinsky A 2020 Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network *Phys. D: Nonlinear Phenom.* **404** 132306
[7] Miao H, Li B, Sun C and Liu J 2019 Joint learning of degradation assessment and RUL prediction for aeroengines via dual-task deep LSTM networks *IEEE Trans. Ind. Inform.* **15** 5023–32
[8] Zheng S, Ristovski K, Farahat A and Gupta C 2017 Long short-term memory network for remaining useful life estimation *2017 IEEE Int. Conf. Prognostics and Health Management ICPHM 2017* pp 88–95

[9] Zhang J, Wang P, Yan R and Gao R X 2018 Long short-term memory for machine remaining life prediction *J. Manuf. Syst.* **48** 78–86

[10] Huang C G, Huang H Z and Li Y F 2019 A bidirectional LSTM prognostics method under multiple operational conditions *IEEE Trans. Ind. Electron.* **66** 8792–802

[11] Cheng Y, Wang C, Wu J, Zhu H and Lee C K M 2022 Multi-dimensional recurrent neural network for remaining useful life prediction under variable operating conditions and multiple fault modes *Appl. Soft Comput.* **118** 108507

[12] Yang Z B, Zhang J P, Zhao Z B, Zhai Z and Chen X F 2020 Interpreting network knowledge with attention mechanism for bearing fault diagnosis *Appl. Soft Comput. J.* **97** 106829

[13] Nguyen R, Singh S K and Rai R 2023 Physics-infused fuzzy generative adversarial network for robust failure prognosis *Mech. Syst. Signal Process.* **184** 109611

[14] Das A, Hussain S, Yang F, Habibullah M S and Kumar A 2019 Deep recurrent architecture with attention for remaining useful life estimation *IEEE Reg. 10 Annual Int. Conf. Proc. /TENCON* (*October 2019*) pp 2093–8

[15] Liu H, Liu Z, Jia W and Lin X 2021 Remaining useful life prediction using a novel feature-attention-based end-to-end approach *IEEE Trans. Ind. Inform.* **17** 1197–207

[16] Sun Z, Wang Y and Gao J 2022 Intelligent fault warning method of rotating machinery with intraclass and interclass infographic embedding *Meas. Sci. Technol.* **33** 114008

[17] Zio E 2022 Prognostics and Health Management (PHM): where are we and where do we (need to) go in theory and practice *Reliab. Eng. Syst. Saf.* **218** 108119

[18] Wei Y, Wu D and Terpenny J 2021 Learning the health index of complex systems using dynamic conditional variational autoencoders *Reliab. Eng. Syst. Saf.* **216** 108004

[19] Li T, Zhao Z, Sun C, Yan R and Chen X 2021 Hierarchical attention graph convolutional network to fuse multi-sensor signals for remaining useful life prediction *Reliab. Eng. Syst. Saf.* **215** 107878

[20] Chen Z, Wu M, Zhao R, Guretno F, Yan R and Li X 2021 Machine remaining useful life prediction via an attention-based deep learning approach *IEEE Trans. Ind. Electron.* **68** 2521–31

[21] Zhao Y and Wang Y 2021 Remaining useful life prediction for multi-sensor systems using a novel end-to-end deep-learning method *Meas. J. Int. Meas. Confed.* **182** 109685

[22] Saxena A, Goebel K, Simon D and Eklund N 2008 Damage propagation modeling for aircraft engine run-to-failure simulation *2008 Int. Conf. Prognostics Health Management PHM 2008* (https://doi.org/10.1109/PHM.2008.4711414)

[23] Li X, Ding Q and Sun J Q 2018 Remaining useful life estimation in prognostics using deep convolution neural networks *Reliab. Eng. Syst. Saf.* **172** 1–11

[24] Chao M A, Kulkarni C, Goebel K and Fink O 2021 Aircraft engine run-to-failure dataset under real flight conditions for prognostics and diagnostics *Data* **6** 1–14

[25] Navathe S B, Wu W, Shekhar S, Du X, Sean Wang X and Xiong H 2016 Deep convolutional neural network based regression approach for estimation of remaining useful life *Lect. Notes Comput. Sci.* **9642** 214–28

[26] Song T, Liu C, Wu R, Jin Y and Jiang D 2022 A hierarchical scheme for remaining useful life prediction with long short-term memory networks *Neurocomputing* **487** 22–33

[27] Arias Chao M, Kulkarni C, Goebel K and Fink O 2022 Fusing physics-based and deep learning models for prognostics *Reliab. Eng. Syst. Saf.* **217** 107961

[28] Mo Y, Wu Q, Li X and Huang B 2021 Remaining useful life estimation via transformer encoder enhanced by a gated convolutional unit *J. Intell. Manuf.* **32** 1997–2006

[29] Liu L, Song X and Zhou Z 2022 Aircraft engine remaining useful life estimation via a double attention-based data-driven architecture *Reliab. Eng. Syst. Saf.* **221** 108330

[30] Costa N and Sánchez L 2022 Variational encoding approach for interpretable assessment of remaining useful life estimation *Reliab. Eng. Syst. Saf.* **222** 108353

[31] Pontes F J, Amorim G F, Balestrassi P P, Paiva A P and Ferreira J R 2016 Design of experiments and focused grid search for neural network parameter optimization *Neurocomputing* **186** 22–34

[32] Xiang S, Qin Y, Liu F and Gryllias K 2022 Automatic multi-differential deep learning and its application to machine remaining useful life prediction *Reliab. Eng. Syst. Saf.* **223** 108531

[33] Greff K, Srivastava R K, Koutnik J, Steunebrink B R and Schmidhuber J 2017 LSTM: a search space odyssey *IEEE Trans. Neural Netw. Learn. Syst.* **28** 2222–32

[34] Al-Dulaimi A, Zabihi S, Asif A and Mohammadi A 2019 A multimodal and hybrid deep neural network model for remaining useful life estimation *Comput. Ind.* **108** 186–96

[35] Xia J, Feng Y, Lu C, Fei C and Xue X 2021 LSTM-based multi-layer self-attention method for remaining useful life estimation of mechanical systems *Eng. Fail. Anal.* **125** 105385

[36] Cheng Y, Hu K, Wu J, Zhu H and Shao X 2022 Autoencoder quasi-recurrent neural networks for remaining useful life prediction of engineering systems *IEEE/ASME Trans. Mechatronics* **27** 1081–92

[37] Tian H, Yang L and Ju B 2023 Spatial correlation and temporal attention-based LSTM for remaining useful life prediction of turbofan engine *Meas. J. Int. Meas. Confed.* **214** 112816

[38] Zhang J, Jiang Y, Wu S, Li X, Luo H and Yin S 2022 Prediction of remaining useful life based on bidirectional gated recurrent unit with temporal self-attention mechanism *Reliab. Eng. Syst. Saf.* **221** 108297

[39] Zhang Z, Song W and Li Q 2022 Dual-aspect self-attention based on transformer for remaining useful life prediction *IEEE Trans. Instrum. Meas.* **71** 1–11

[40] Li H, Cao P, Wang X, Yi B, Huang M, Sun Q and Zhang Y 2023 Multi-task spatio-temporal augmented net for industry equipment remaining useful life prediction *Adv. Eng. Inform.* **55** 101898

[41] Li Y, Chen Y, Hu Z and Zhang H 2023 Remaining useful life prediction of aero-engine enabled by fusing knowledge and deep learning models *Reliab. Eng. Syst. Saf.* **229** 108869

[42] Zhao K, Jia Z, Jia F and Shao H 2023 Multi-scale integrated deep self-attention network for predicting remaining useful life of aero-engine *Eng. Appl. Artif. Intell.* **120** 105860

[43] Biggio L, Wieland A, Chao M A, Kastanis I and Fink O 2021 Uncertainty-aware prognosis via deep gaussian process *IEEE Access* **9** 123517–27

[44] Ren L, Wang H and Huang G 2023 DLformer: a dynamic length transformer-based network for efficient feature representation in remaining useful life prediction *IEEE Trans. Neural Netw. Learn. Syst.* 1–11

[45] Tseng S H and Tran K D 2023 Predicting maintenance through an attention long short-term memory projected model *J. Intell. Manuf.* **35** 807–24