

## Research Article

# A Spatio-Temporal Attention Mechanism Based Approach for Remaining Useful Life Prediction of Turbofan Engine

Cheng Peng <sup>1,2</sup>, Jiaqi Wu <sup>1</sup>, Zhaohui Tang <sup>2</sup>, Xinpan Yuan,<sup>1</sup> and Changyun Li <sup>1</sup>

<sup>1</sup>School of Computer Science, Hunan University of Technology, Zhuzhou 412000, China

<sup>2</sup>School of Automation, Central South University, Changsha 418003, China

Correspondence should be addressed to Changyun Li; [changyunli2022@163.com](mailto:changyunli2022@163.com)

Received 20 July 2022; Revised 4 October 2022; Accepted 7 October 2022; Published 14 October 2022

Academic Editor: Tapan Senapati

Copyright © 2022 Cheng Peng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The time-series data generated by turbofan engines has a great degree of complexity and dynamics. At present, recurrent neural networks are commonly used to model and forecast the remaining useful life (RUL). The relationship of the sample data is not taken into account, and there are issues such as gradient explosion. In view of this, a spatio-temporal attention model is proposed, which comprehensively relates to the temporal association of data features and the hidden state of data features in space. At the same time, position coding is performed on the temporal relationship, avoiding the use of recurrent neural networks. Experimental results show that by combining the two dimensions, the predictive performance of the model is significantly improved. Compared with different methods on the four data sets of the commercial modular aerospace propulsion system simulation (C-MAPSS), the stability and prediction accuracy of the spatio-temporal attention model are better than that of alternative methods.

## 1. Introduction

Among the functions of failure prediction and health management, RUL prediction is the core function. The turbofan engine is one of the key power equipment of the aircraft, and its safe and stable work is extremely crucial. When the accuracy of the turbofan engine RUL prediction reaches a certain level, the original regular maintenance can be turned into predictive maintenance, which will increase the overall use time of the equipment, avoid irreparable deterioration of the equipment, and reduce the expenditure of human and material resources. Therefore, it is extremely essential to improve the RUL prediction accuracy of turbofan engines.

At present, as a data-driven method, deep learning is widely used for RUL prediction. Regarding the research on RUL prediction of turbofan engines, deep learning methods can be divided into methods based on recurrent neural networks [1], convolutional neural networks [2], and hybrid model methods.

The methods based on the recurrent neural network are to use the recurrent neural network to process the time

relationship of the sequence data and predict the RUL of the turbofan engine. In order to improve the accuracy of RUL prediction, Heimes [3] proposed a prediction method based on recurrent neural network (RNN), which uses time gradient calculation and extends Kalman filter for training, and predicts the RUL of turbofan engines. However, RNN has the problem of gradient disappearance. In order to predict RUL stably in an environment of complex operation, mixed faults, and strong noise, Yuan et al. [4] established a long short-term memory (LSTM) neural network model for fault diagnosis and RUL prediction of turbofan engines. But LSTM can only infer the feature relationship from one direction of the time series. In order to discover the long-term dependence relationship of sensor time series signals, Wu et al. [5] designed an RUL prediction method based on deep long short-term memory (DLSTM) network using multiple sensor time series signals. However, DLSTM has the problem of long calculation time. In order to solve the problem, the generated vector contribution of each time step pair of RNN on the autoencoder is equal, Duan et al. [6] adopted a method of assigning weight to each time step through an attention mechanism to improve the prediction

ability of BiGRU. However, the model does not consider the relationship between the sensor's positions in space. Aiming at the problem of complex features of sensor data in RUL prediction, Muneer et al. [7] applied an LSTM model based on the attention mechanism to achieve accurate RUL prediction. But recurrent neural networks cannot achieve parallel computing. A compromise in performance against computational efficiency is found to be unfavorable to meeting practical needs [8].

The methods based on the convolutional neural network are to use the convolutional neural network to extract the local features of the sequence data and then predict the RUL. Wen et al. [9] established a new residual convolutional neural network (ResCNN) to solve the problem of gradient disappearance and predict the RUL of turbofan engines but did not consider the relationship between the sensors at each time step. In order to solve the problem of model adaptation to different data sources, Li et al. [10] proposed a domain adaptive RUL prediction by integrating adaptive batch normalization (AdaBN) into a deep convolutional neural network (DCNN) model, but mapping the features of different data sources to the same feature relationship space is still a problem. In view of the limited ability of traditional data-driven methods in extracting complex features, Li et al. [11] designed a multiscale convolutional neural network (MSCNN) to directly establish the relationship between monitoring data and RUL, but the size of the multiscale convolution kernel is determined empirically.

The hybrid model method combines the advantages of several methods. In order to make full use of the advantages of CNN and LSTM models, Borst [12] proposed a method combining LSTM and CNN to predict the RUL of turbine engines. Remadna et al. [13] use CNN with bi-directional long short-term memory (BDLSTM) networks where CNN extract spatial features while BDLSTM extracts temporal features. Chen et al. [14] combines the DCNN and the BDLSTM was used to train the source domain and target domain data simultaneously to extract the common features of the equipment under the condition of sufficient samples. Jiang et al. [15] propose a data-driven method based on BDLSTM and multiscale convolutional neural network (MSCNN) for RUL estimation. But how to effectively splice CNN and LSTM models is currently a problem that needs to be resolved.

In conclusion, a lot of research has made great progress on the RUL prediction of turbofan engines. However, the neural network still has the following problems in RUL prediction. Firstly, the recurrent neural network can only infer the internal information of the time series from one direction or two directions, but does not consider the relationship between each time tick, and cannot achieve parallel operations. Secondly, for observational data, most studies only consider how to predict RUL in terms of time while ignoring the spatial connection of data features that also assists in the prediction of RUL. Last but not least, for long-term sequences, the recurrent neural network has the problem of gradient disappearance or gradient explosion. Therefore, how to avoid the gradient problem while also taking into account the position of the time series is also a problem.

In view of the aforementioned shortcomings, this paper proposes a spatio-temporal attention model to predict RUL. The main contributions are as follows. First, from the perspective of time relationships, a time position coding method is designed to code the position of the time series and actively apply the time series relationship to the model. Secondly, in order to overcome the problem of incomplete information on the time dimension and the inability of parallel computing of recurrent neural network, an attention mechanism is used to extract the state relationship of features in the two dimensions of time and space, respectively. Finally, a convolution module is designed, which combines the time feature map and the space feature map, and uses two fully connected layers to predict the RUL of the turbofan engine. All in all, the attention mechanism solves the problem that recurrent neural networks cannot perform parallel computation, and the model does not take into account the spatial relationship of features or the different weights of each timestamp. In order to verify the effectiveness of the method, we made predictions on 4 datasets of NASA's C-MAPSS.

The rest of the paper is organized as follows. The second part introduces the methodology of spatio-temporal attention, including data preprocessing, and spatio-temporal attention models; the third part analyzes the experimental results.

## 2. Materials and Methods

*2.1. The Procedure of Remaining Useful Life Prediction.* It is shown in Figure 1 that the procedure of RUL prediction includes the following 4 steps.

- (1) Data preprocessing: the data in the original data set contains redundant features, and the unit scales between the data are inconsistent, so it is necessary to preprocess the data and then standardize it.
- (2) Data set division: the whole training set is divided into a training set and verification set by early stop strategy. The training set is used to adjust the parameters of the model during model training; the verification set is used to train the early stop model. That is, in 8 training cycles, if the overall RMSE of the verification set does not decline, the training will be terminated.
- (3) Model Construction: build a spatio-temporal attention mechanism model, and in the processing time dimension, first encode the input data, and then use it as the input of the model.
- (4) RUL prediction: use the trained model to predict the remaining life on the test set and evaluate the corresponding indicators.

The structure of the spatial-temporal attention mechanism model is shown in Figure 2. As can be seen from the diagram, the whole structure can be divided into spatial attention mechanism, temporal attention mechanism, convolution layer fusion, and full connection layer. Spatial attention mechanism and temporal attention mechanism

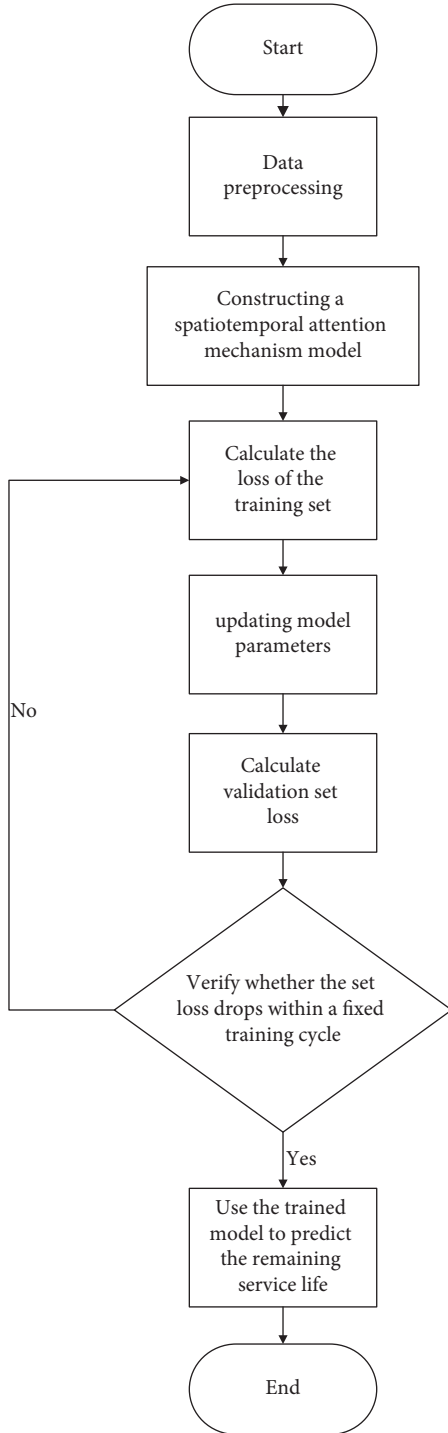


FIGURE 1: RUL forecast flow chart.

modules obtain the spatial and temporal characteristics of the sequence data mainly through the multi-head self-attention mechanism [16]. Convolution layer fusion further extracts features, fuses the feature map into a vector, and reduces the feature dimension. The fully connected layer predicts RUL by eigenvectors. The process and principles of this approach are described in detail below.

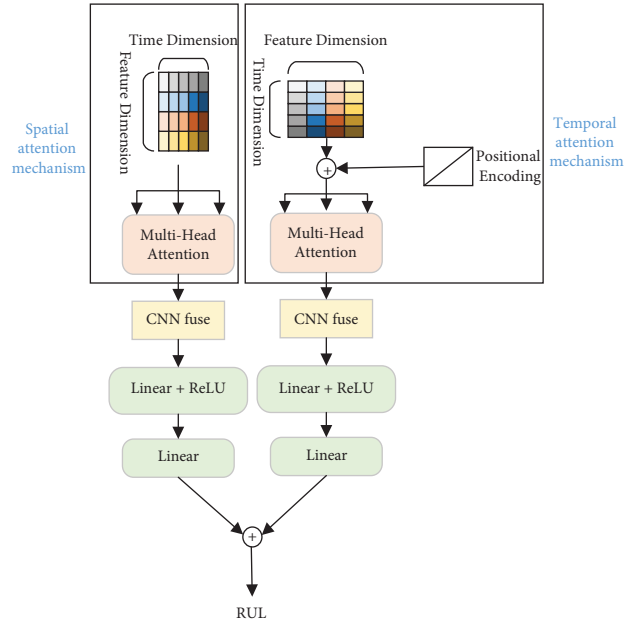


FIGURE 2: Spatial-temporal attention mechanism model structure.

**2.2. Data Preprocessing.** The C-MAPSS dataset contains three operational settings and 21 sensor data. Figures 3–6 show the standardized 24 index data of Engine Unit 1 of the FD001, FD002, FD003, and FD004 sub-datasets. In these figures, each color represents a feature; RUL stands for the numerical value of abscissa; the ordinate value indicates the magnitude of the characteristic value. The model is able to learn the functional relationship between RUL and characteristic value.

However, some of the data features do not change much over the entire time period, which is not helpful for the prediction of the model. Therefore, it is necessary to select data features that vary over time to predict RUL. A large number of studies have shown that the prognosability method can be used to represent the variability of features. This paper uses the prognosability [17] method to select features. As in the following formula:

$$\text{prognosability} = \exp\left(-\frac{\text{std}_j(x_j(N_j))}{\text{mean}_j|x_j(1) - x_j(N_j)|}\right), j = 1, \dots, M. \quad (1)$$

If a feature's prognosability is equal to 0 or NaN, it is not preserved. Conversely, the autonomy of the model has been improved, and the problem of relying on human experts for feature selection has been addressed [8].

After selecting features, the data needs to be standardized and scaled to the same scale. Z-score [18] can unify data of different magnitudes into the same magnitude, and the conversion process can easily extend to more complex datasets, so the Z-score method is used to standardize the data. As in the following formula:

$$z = \frac{x - u}{\sigma}. \quad (2)$$

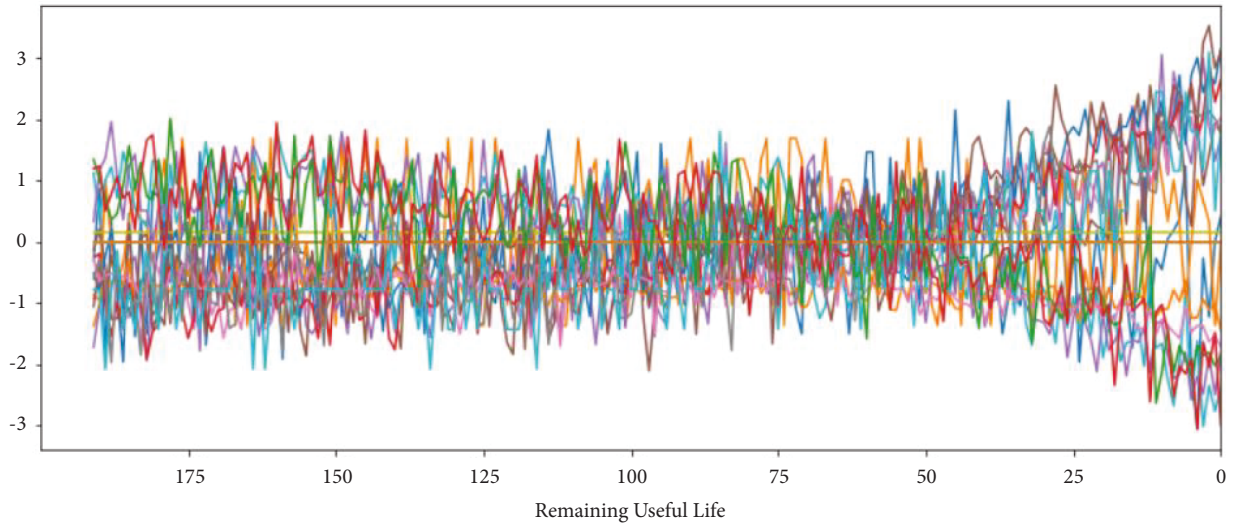


FIGURE 3: Engine unit 1 sensor data in FD001 sub-dataset.

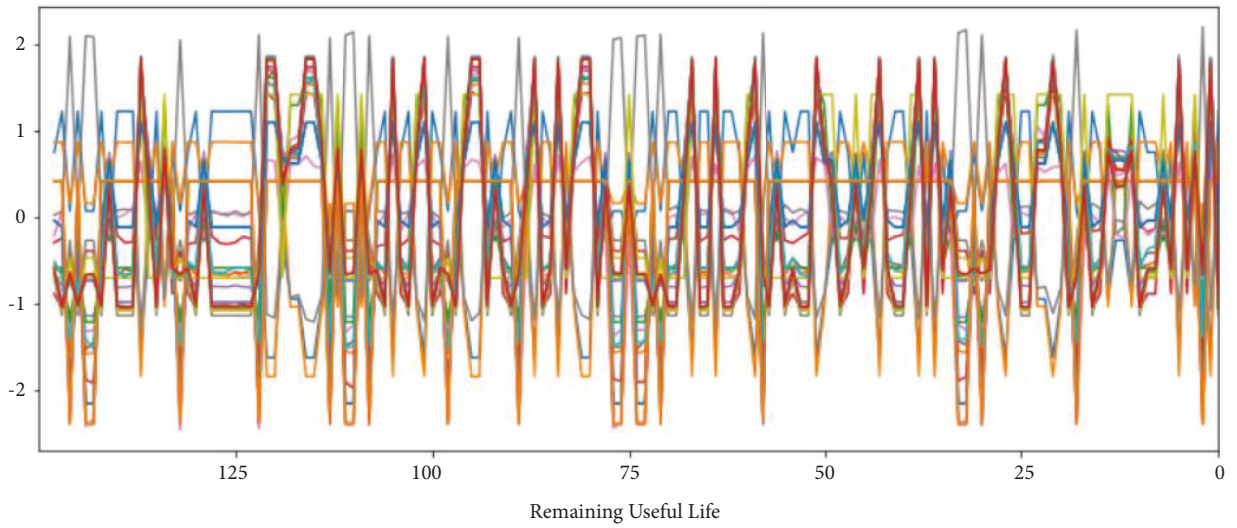


FIGURE 4: Engine unit 1 sensor data in FD002 sub-dataset.

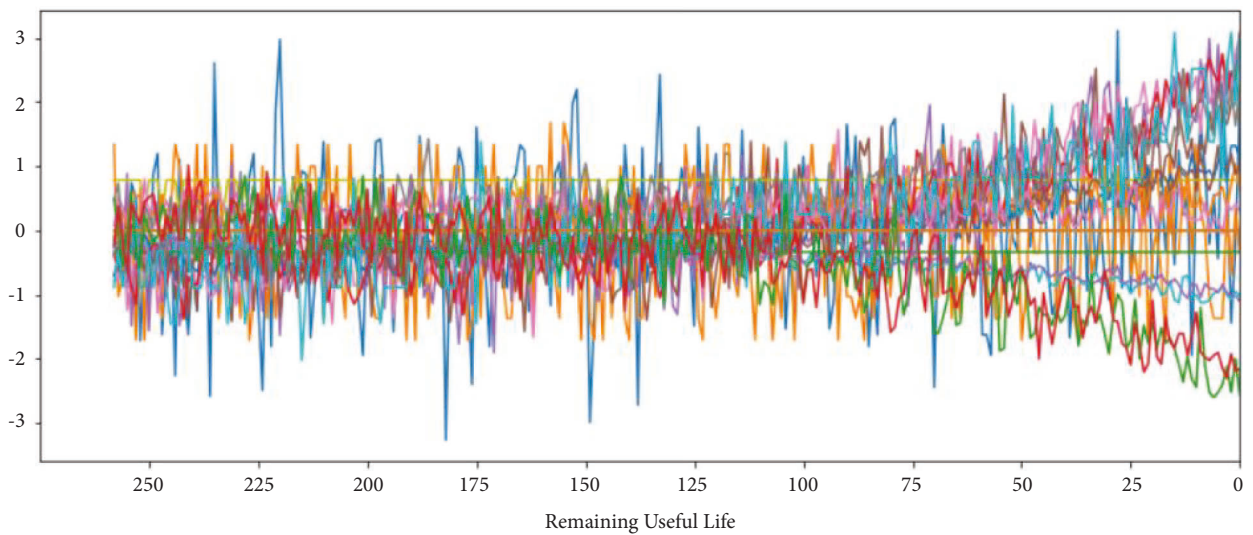


FIGURE 5: Engine unit 1 sensor data in FD003 sub-dataset.

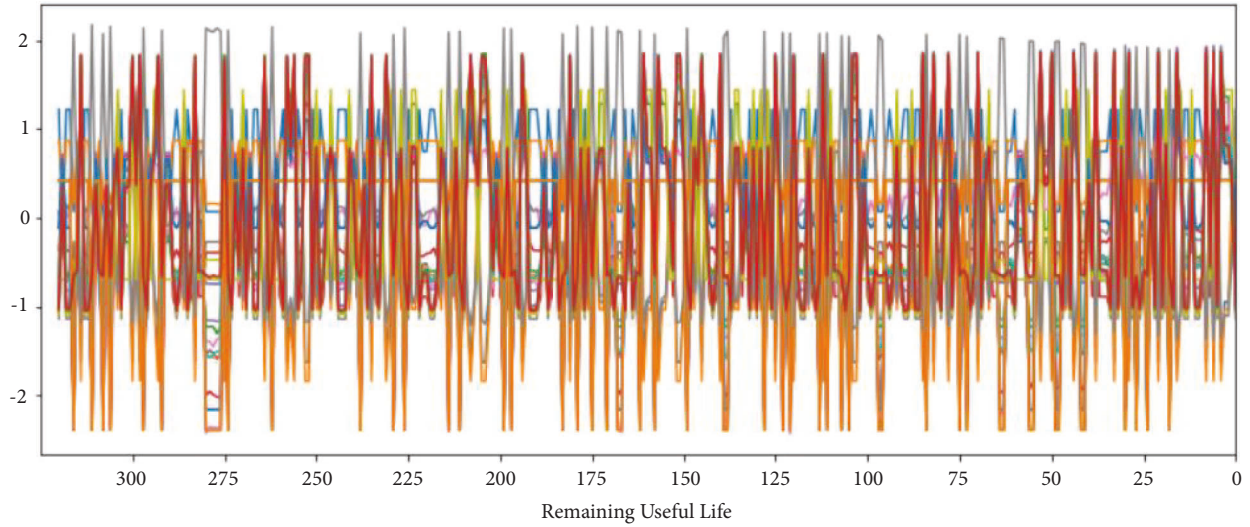


FIGURE 6: Engine unit 1 sensor data in FD004 sub-dataset.

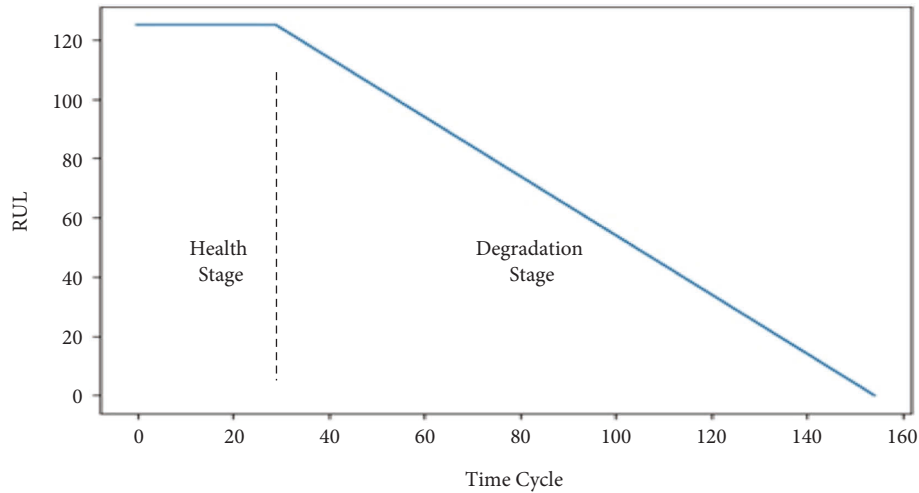


FIGURE 7: Distribution of turbo engine real life.

$\mu$  is the mean of the selected feature, the standard deviation of the selected feature, and  $x$  represents the feature value at a given time.

The health state of a turbofan engine is generally divided into health state and deterioration state. In a healthy state, the RUL of a turbofan engine is in a constant state. In the degenerate state, its RUL is in a descending state. Therefore, a health threshold needs to be set for turbo engine equipment. The results show that when the initial RUL value in the C-MAPSS dataset is in the range 120–130, the output is stable. Therefore, we set the RUL threshold to 125. See Figure 7 for details.

After these processes, the data also needs to be divided into time windows as input to the model. The input of the model  $= [x_1, x_2, \dots, x_f]$ , where,

$$x_i = \begin{bmatrix} x_i^{window\_start} \\ x_i^{window\_start+1} \\ \vdots \\ x_i^{window\_end} \end{bmatrix}, i = 1, \dots, f. \quad (3)$$

$f$  is the number of time feature selected, (window\_end-window\_start) is the size of the time window, and the time step is set to 1 for more training samples.

**2.3. Spatial-Temporal Feature Extraction.** Most current studies only consider how to predict RUL in terms of temporal relationships, ignoring the spatial relationships of data. This paper considers the internal relationships between

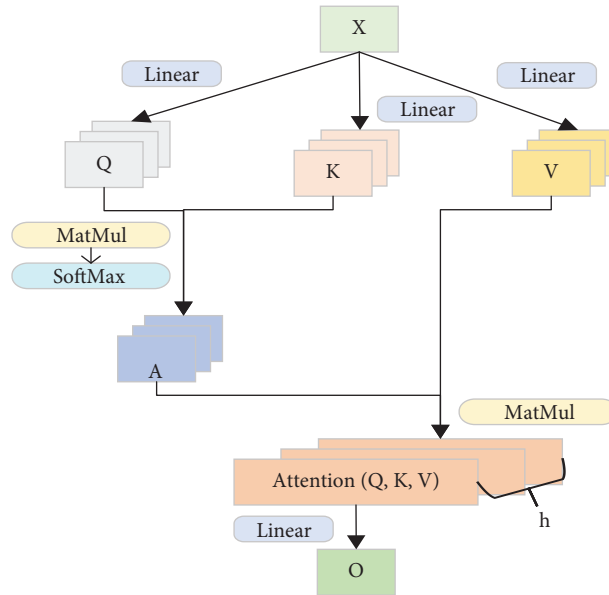


FIGURE 8: Multi-head attention structure.

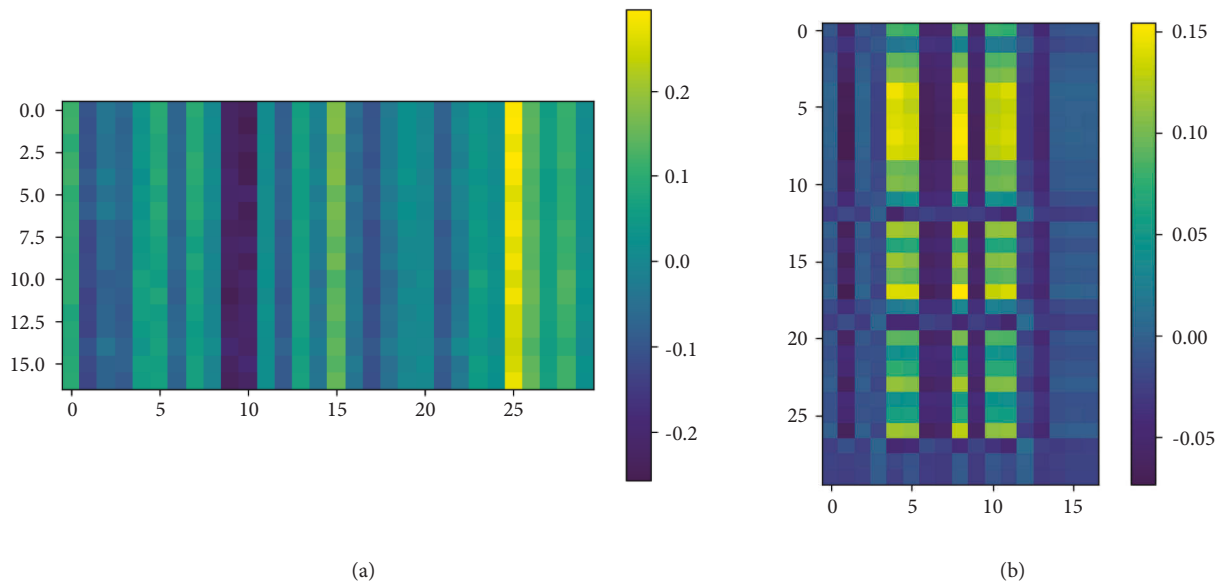


FIGURE 9: Spatio-temporal characteristic map; (a) spatial feature map; (b) time characteristic diagram.

extracting data in both temporal and spatial directions. However, data may have multiple relationships in the same direction. In this paper, multi-head attention [16] is used to extract multiple connections of data in both time and space dimensions while avoiding the problem of gradient disappearance or gradient explosion of circulating nerves and

improving the speed of parallel computing. The multi-head attention structure is shown in Figure 8.

Where  $X$  represents input data;  $A$  represents the relationship weight between the row vectors in the input data  $x$ , which is derived from the matrices  $Q$  and  $K$ ;  $V$  represents the specific real value characteristics of the row

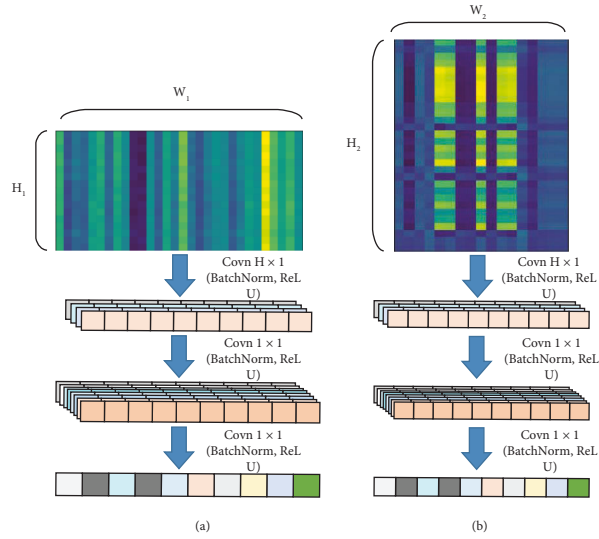


FIGURE 10: Convolution fusion structure. (a) CNN in spatial feature map. (b) CNN in time feature map.

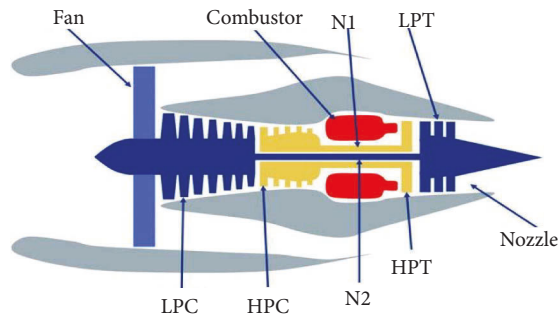


FIGURE 11: Engine structure diagram.

TABLE 1: Data set introduction.

| Sub-datasets                         | FD001                 | FD002                 | FD003                                  | FD004                                  |
|--------------------------------------|-----------------------|-----------------------|--|--|
| Engine units in the training dataset | 100                   | 260                   | 100                                    | 249                                    |
| Engine units in the test dataset     | 100                   | 259                   | 100                                    | 248                                    |
| Fault modes                          | One (HPC degradation) | One (HPC degradation) | Two (HPC degradation, fan degradation) | Two (HPC degradation, fan degradation) |
| Conditions                           | One (sea level)       | Six                   | One (sea level)                        | Six                                    |
| Training samples                     | 17731                 | 48819                 | 21820                                  | 57522                                  |
| Test samples                         | 100                   | 259                   | 100                                    | 248                                    |

TABLE 2: Sensor introduction.

| Index | Description                     | Symbol  |
|-------|---------------------------------|---------|
| 1     | Total temperature at fan inlet  | °R      |
| 2     | Total temperature at LPC outlet | °R      |
| 3     | Total temperature at HPC outlet | °R      |
| 4     | Total temperature at LPT outlet | °R      |
| 5     | Pressure at fan inlet           | psia    |
| 6     | Total pressure in bypass-duct   | psia    |
| 7     | Total pressure at HPC outlet    | psia    |
| 8     | Physical fan speed              | rpm     |
| 9     | Physical core speed             | rpm     |
| 10    | Engine pressure ratio (P50/P2)  | —       |
| 11    | Static pressure at HPC outlet   | psia    |
| 12    | Ratio of fuel flow to Ps30      | Pps/psi |
| 13    | Corrected fan speed             | rpm     |
| 14    | Corrected core speed            | rpm     |
| 15    | Bypass ratio                    | —       |
| 16    | Burner fuel-air ratio           | —       |
| 17    | Bleed enthalpy                  | —       |
| 18    | Demanded fan speed              | rpm     |
| 19    | Demanded corrected fan speed    | rpm     |
| 20    | HPT coolant bleed               | lbm/s   |
| 21    | LPT coolant bleed               | lbm/s   |

vector in the input data  $X$ ;  $h$  represents the number of relationships. The calculation formula of multi-head attention is as follows.

$$\begin{aligned}
 Q &= XW^q, \\
 V &= XW^v, \\
 K &= XW^k, \\
 A &= \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right),
 \end{aligned} \tag{4}$$

$$\text{Attention}(Q, K, V) = AV.$$

When using the attention mechanism method, some current studies usually use the attention mechanism to extract the relationship between the time of the vector, such as the relationship between  $t_1$  time vector and  $t_2$  time vector. Then the time relationship is input into the recurrent neural network to predict RUL, but the recurrent neural network is prone to the problem of gradient disappearance or gradient explosion. If only the attention mechanism is used, the model may ignore the position of the time series. For example, it does not consider that the vector at  $t_1$  is before the vector at  $t_2$ . Therefore, this paper encodes the position of the original data to solve the problem that the attention mechanism does not consider the time sequence.

Position coding is a method of quadratic representation of each time slot in the sequence with the position information of different time slots. As mentioned above, the self-attention mechanism is to extract the relationship between vectors at different times. It does not have the ability to learn time series information like recurrent neural network, so it needs to encode the position and actively input the time series information to the model.

Location coding is a new representation vector that combines the information of time and the original time series. In this way, the model can learn the time position information of the original time series. The proposed position coding formula is as follows.

$$\begin{aligned}
 \text{encoding} &= \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ \vdots \\ e_h \end{bmatrix}, \\
 e_i &= [e_i^1 \cdots e_i^f], \\
 e_i^j &= e_i^j, \{j = 1, 2, \dots, f\}, \\
 e_i^1 &= \frac{i - \mu}{\sigma}, \\
 \mu &= \frac{1}{h} \sum_{i=1}^h i, \\
 \sigma &= \sqrt{\frac{1}{h} \sum_{i=1}^h (i - \mu)^2}.
 \end{aligned} \tag{5}$$

Encoding represents the coding information of the whole input time window,  $H$  represents the length of the time series, and  $f$  represents the number of selected features;  $e_i$  represents the coding information vector at the  $i$ -th time in



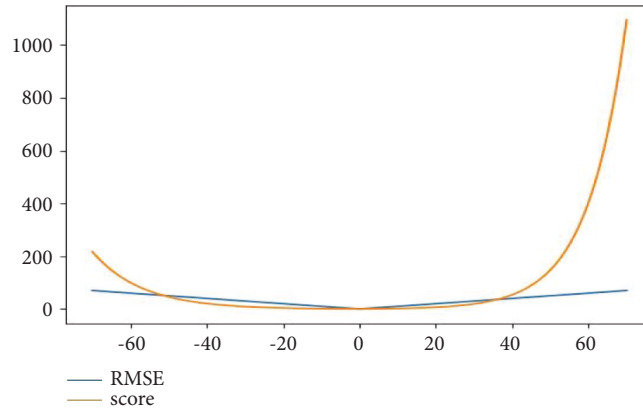


FIGURE 12: Evaluation index function.

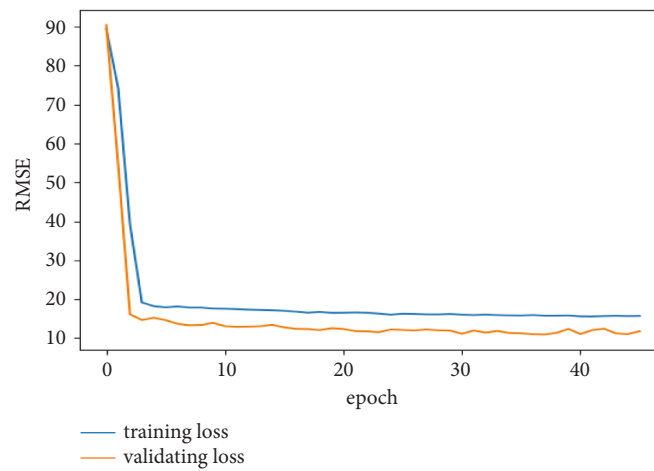


FIGURE 13: Training loss map.

TABLE 3: Experimental results of different training strategies.

| Method         | FD001 training set |               | FD001 test set |               |
|----------------|--------------------|---------------|----------------|---------------|
|                | RMSE               | Score         | RMSE           | Score         |
| Early stopping | 10.89              | 371.14        | <b>11.07</b>   | <b>312.55</b> |
| 200 epochs     | <b>8.44</b>        | <b>202.26</b> | 13.151         | 374.93        |

TABLE 4: Comparison of experimental results.

| Method                    | FD001        |               | FD002        |               | FD003        |               | FD004        |                |
|---------------------------|--------------|---------------|--------------|---------------|--------------|---------------|--------------|----------------|
|                           | RMSE         | Score         | RMSE         | Score         | RMSE         | Score         | RMSE         | Score          |
| Spatio-temporal attention | <b>11.07</b> | 312.55        | <b>18.10</b> | <b>1339.5</b> | <b>10.73</b> | 329.95        | <b>17.00</b> | <b>2476.90</b> |
| BiGRU-AS [6]              | 13.68        | 284           | 20.81        | 2454          | 15.53        | 428           | 27.31        | 4708           |
| Ensemble ResCNN [9]       | 12.16        | 212.48        | 20.85        | 2087.77       | 12.01        | <b>180.76</b> | 24.97        | 3400.44        |
| AdaBN-DCNN [10]           | 13.17        | 279           | 20.87        | 2020          | 14.97        | 817           | 24.57        | 3690           |
| MS-DCNN [11]              | 11.44        | <b>196.22</b> | 19.35        | 3747          | 11.67        | 241.89        | 22.22        | 4844           |
| Semi-supervised [23]      | 12.56        | 231           | 22.73        | 3366          | 12.1         | 251           | 22.66        | 2840           |
| DCNN [24]                 | 12.61        | 273.7         | 22.36        | 10412         | 12.64        | 284.1         | 23.31        | 12466          |
| MODBNE [25]               | 15.04        | 334.23        | 25.05        | 5585.34       | 12.51        | 421.91        | 28.66        | 6557.62        |
| DBN [26]                  | 15.21        | 417.59        | 27.12        | 9031.64       | 14.71        | 442.43        | 29.88        | 7954.51        |
| LSTM [25]                 | 16.14        | 338           | 24.49        | 4450          | 16.18        | 852           | 28.17        | 5550           |
| MLP [26]                  | 16.78        | 560.59        | 28.78        | 14026.72      | 18.47        | 479.85        | 30.96        | 10444.35       |

FD001

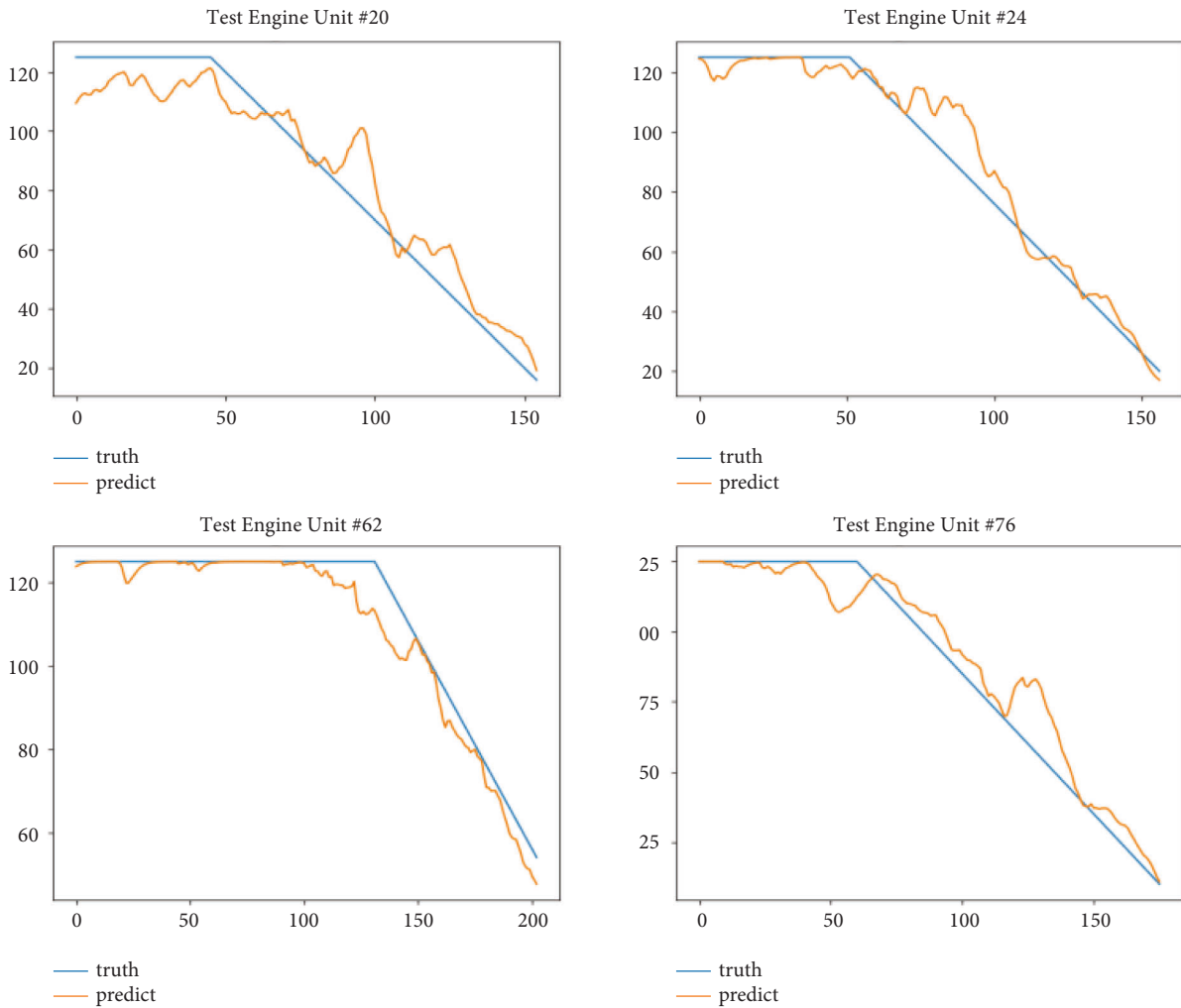


FIGURE 14: Sub-data set FD001 prediction RUL curve.

the input time window,  $\mu$  represents the expectation of the length of the time series, and  $\sigma$  represents the standard deviation of the length of the time series.

**2.4. Convolution Fusion.** After the original data is processed by the attention mechanism, the characteristic diagram of the data in time and space is obtained, as shown in Figure 9. Each line vector on the spatial feature graph represents the spatial state relationship between the original input line feature and all features; each line vector on the time feature graph represents the correlation between the features of each time moment and the features in the whole-time window. For the feature map, the convolution neural network is used to reduce the dimension and further extract the internal information of the data. Then RUL is predicted by two-layer fully connected neural network.

In this paper, the feature map is convoluted in only one direction, so the feature vector can be fused into a specific

value. Therefore, the high  $h$  of the feature map is reduced to 1, which plays a role in reducing the dimension. We call this convolution fusion, and Figure 10 shows the structure of the convolution fusion model.

### 3. Results and Discussion

**3.1. Data Set Introduction.** The data set used in the experiment is the C-MAPSS data set of NASA [19]. This data set is generated by the simulation software simulating the aero-engine. The schematic diagram of the simulated engine is shown in Figure 11, including high-pressure turbofan, high-pressure compressor, fan, nozzle, and burner. See Table 1 for a detailed description of the data set.

Table 1 shows that the data set is divided into four sub-data sets FD001, FD002, FD003, and FD004 according to different operating conditions and fault modes. Each sub-data is divided into training data set and test data set, which records three operation settings and 21 sensor data of scroll

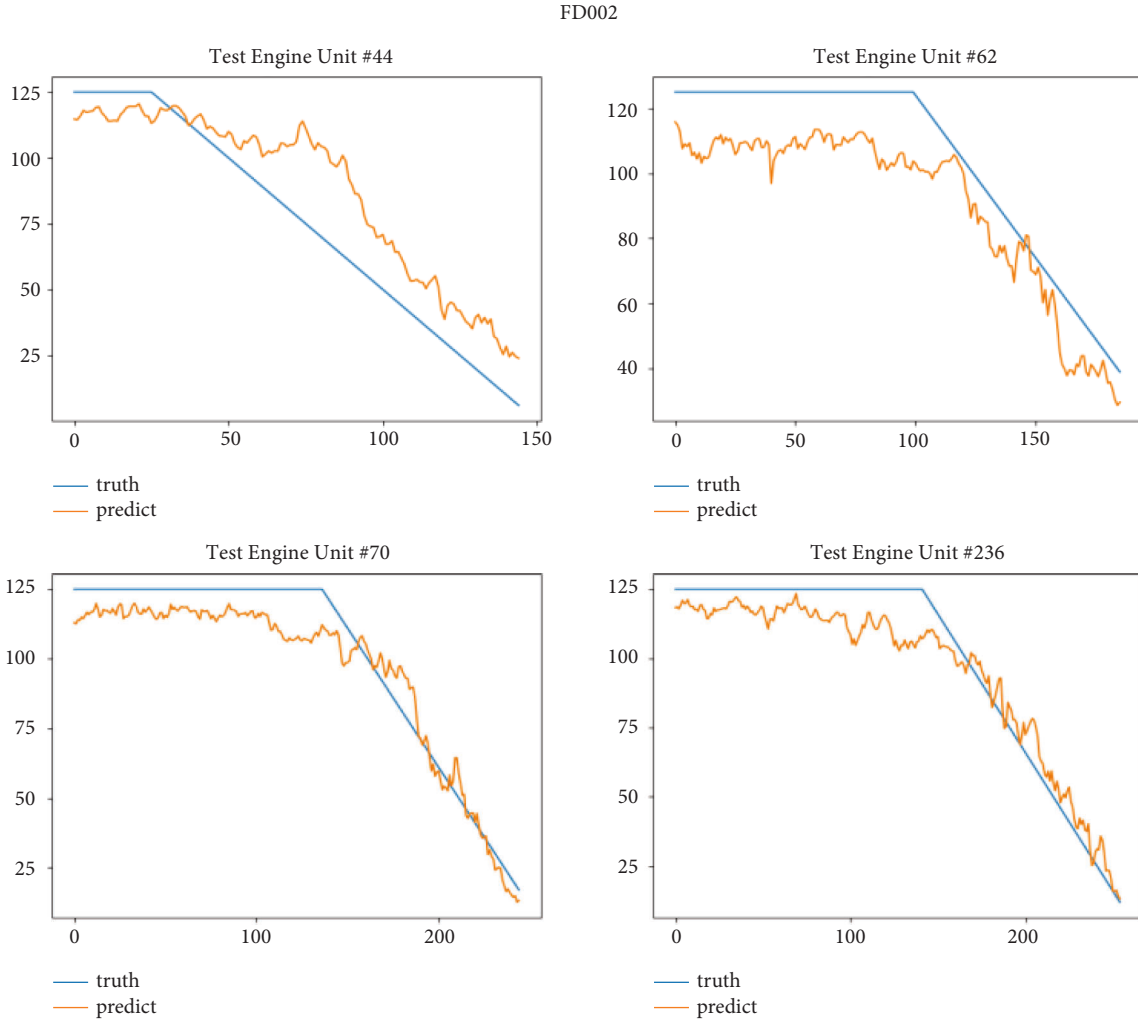


FIGURE 15: RUL curve of sub-dataset FD002 prediction.

engine unit at each time. We use these indicators to divide spatio-temporal data. FD001 and FD002 sub-datasets contain one failure mode (HPC degradation), and FD003 and FD004 contain two degradation modes (HPC degradation and fan degradation); FD001 and FD003 have only one operating condition, and FD002 and FD004 have six operating conditions. Because the operation environment of FD002 and FD004 sub-dataset engine unit is complex and changeable, the prediction of RUL of FD002 and FD004 sub-dataset is more difficult. Table 2 [20] lists the 21 sensors that monitor engine conditions.

**3.2. Evaluating Indicator.** In terms of remaining life prediction of the turbofan engine, many important journals [3, 7] references and the International Conference on fault prediction and health management [19] take score and root mean square error (RMSE) as the evaluation indexes of RUL prediction results. In order to enhance comparability, these two evaluation indexes are also used in this paper. The scoring function will have a larger penalty score for the predicted value with larger deviation; the root mean square error considers the real deviation

between the predicted value and the actual value. The specific formula is as follows.

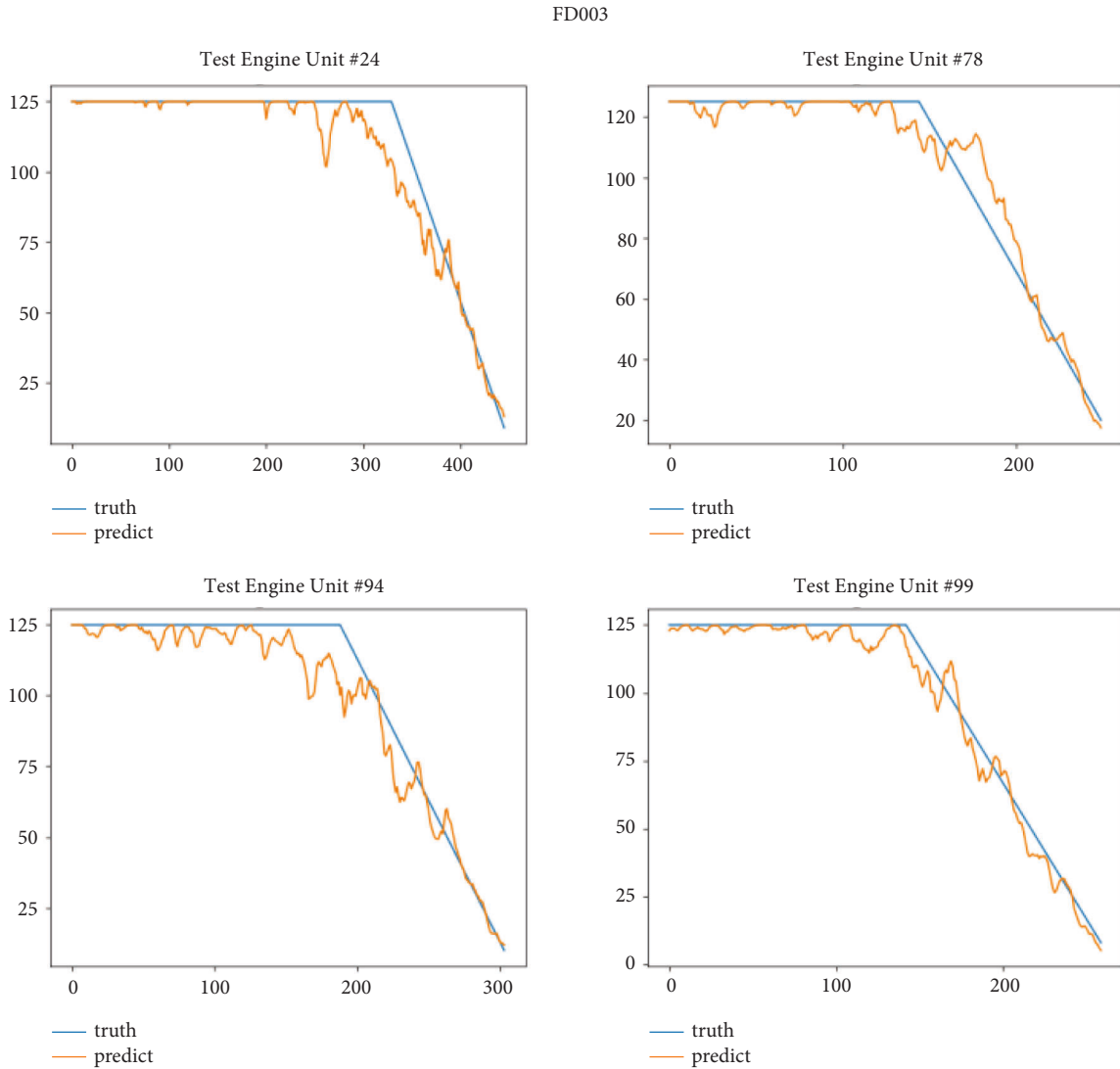
$$s = \sum_{i=1}^N s_i, s_i = \begin{cases} \frac{d_i}{13} - 1, & \text{for } d_i < 0 \\ \frac{d_i}{e^{10}} - 1, & \text{for } d_i \geq 0 \end{cases}, \quad (6)$$

$$d_i = \text{predict}_i - \text{RUL}_i,$$

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

$\text{predict}_i$  represents the predicted value,  $\text{RUL}_i$  represents the real RUL value, and  $N$  represents the number of sample data.

In Figure 12 shows the relationship between RMSE and score; the abscissa value is the difference between the predicted value and the real value, and the ordinate is the specific value of RMSE and score. It can be seen that when the deviation between the predicted value and the real value is large, the value of score increases exponentially.



**3.3. Experiment and Result Analysis.** In order to prevent over fitting of the model, we use dropout [21, 22] technology and verification set early stop strategy. The loss value in the training process is shown in Figure 13.

From Figure 13, it can be seen that the training of more than 40 epochs stopped after the model training. Because the use of early stop strategy avoids the data characteristics in the model over learning training set. At the same time, it can also be seen that during the training process, the RMSE of the verification set is less than that of the training set. Because the dropout technology is added to the full-connection layer. During training, some neural units of the full-connection layer will be inactivated, and when solving the RMSE of the verification set, all the inactivated neural units will be activated. In order to verify the impact of adopting the early stop strategy, we also trained 200 epochs for the model. The experimental results are shown in Table 3.

Table 3 shows that with the early stop strategy, the deviation between the RMSE of the FD001 training set and the FD001 test set is not large, and there is no fitting

phenomenon. However, after 200 cycles of training the model directly, there is a large deviation between the RMSE of the FD001 training set and the FD001 test set, and there is a problem of overfitting. At the same time, the RMSE of the training model with early stop strategy is 10.89 on the FD001 training set, and the RMSE of the model with 200 training cycles is 8.44. The difference is in a reasonable range, and there is no obvious underfitting phenomenon. The prediction effect of the test set of the remaining three sub-data sets is shown in Table 4.

After the model is trained, we test it on the test set of four sub-data sets. This paper selects some engine units with two health stages for display. The FD001 sub-data set selects engine units 20, 24, 62, and 76; FD002 sub-data set selects 44, 62, 70 and 236 engine units; the FD003 sub-data set selects engine units 24, 78, 94 and 99; the FD004 sub-data set selects 213, 235, 244 and 248 engine units. Figures 11–14 show the prediction results of these engine units.

It can be seen from Figures 14 to 17 that the predicted RUL curve presents a piecewise linear state, which truly

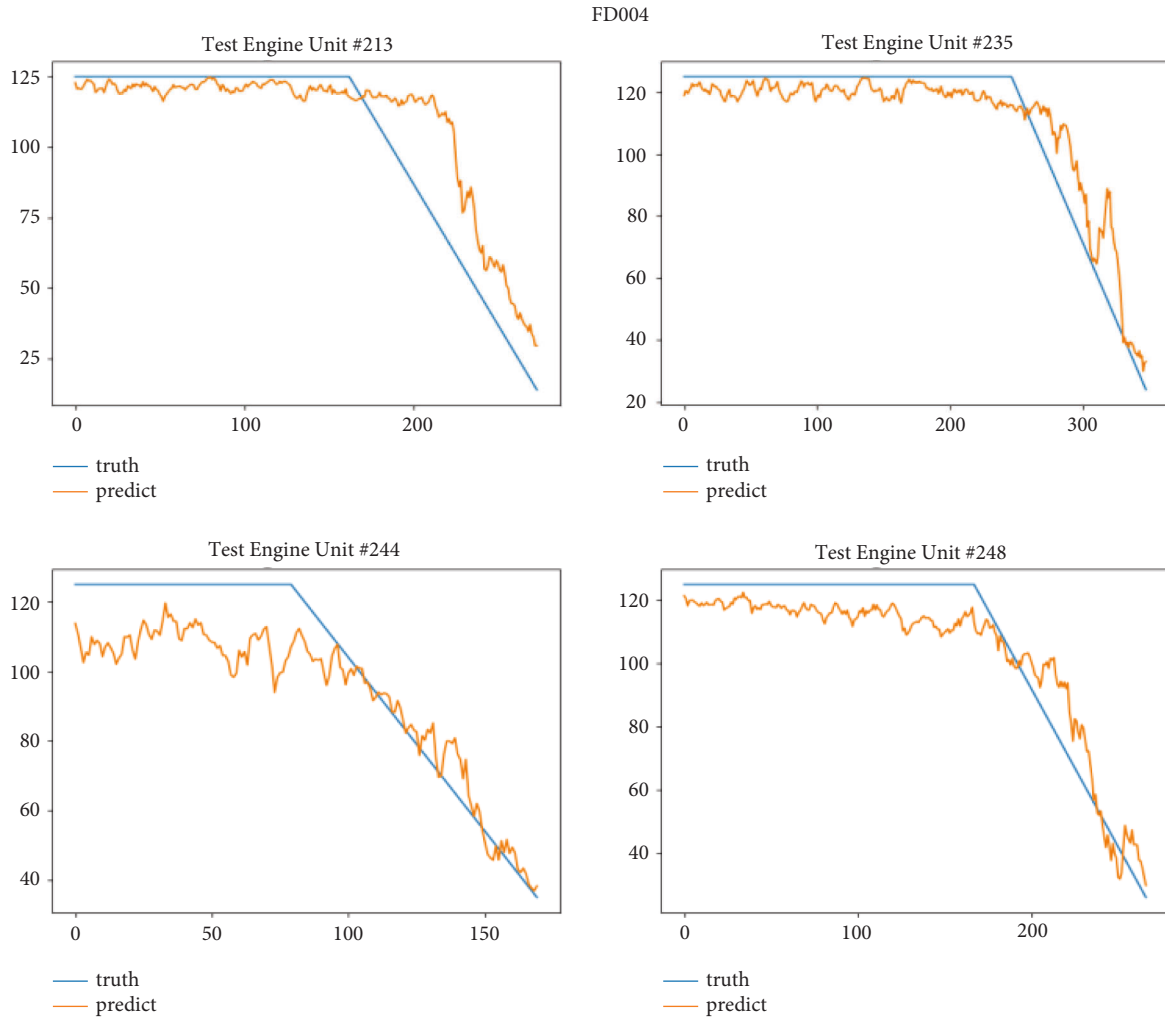


FIGURE 17: RUL prediction data set FD004 curve.

reflects the degradation state of the real RUL. However, since FD001 and FD002 sub-datasets contain one failure mode (HPC degradation), FD003 and FD004 contain two degradation modes (HPC degradation and fan degradation); FD001 and FD003 have only one operating condition, and FD002 and FD004 have six operating conditions. The data distribution of FD001 and FD003 sub-datasets will be relatively regular. See Figures 3–6 for details. Therefore, the prediction accuracy of the model in the FD0001 and FD003 sub-datasets is higher than that in FD0002 and FD004 sub-datasets.

In order to verify the stable prediction accuracy of our proposed method, this paper tests the proposed model on the test set on each sub-data set, and compares the results with the prediction RUL method related to deep learning in recent years. See Table 4 for detailed comparison data.

As shown in Table 4, the spatio-temporal attention method proposed in this paper is superior to most methods in terms of RMSE and score. In the sub-data sets FD002 and FD004 under six different flight conditions, the spatio-temporal attention method reduces the RMSE to 18.10 and 17.00, and the score to 1339.59 and 2476.906, respectively.

This reflects that the proposed spatio-temporal attention method still performs well and has strong robustness in the multi-traffic environment. Compared to the BiGRU-AS method, the spatio-temporal attention takes into account not only the relationship between time series but also the hidden states between sensors in spatial locations, which makes full-use of the data. At the same time, our method actively applies time series weights to the model, which can reduce the effect of data noise. Generally speaking, the spatio-temporal attention method has certain advantages over other methods in the table.

#### 4. Conclusion

In this paper, the proposed attention mechanism model is used to learn the relationship between time series and the relationship between sensors in spatial position, and the RUL of a turbofan engine is predicted by convolution neural network fusion. All in all, this model addresses the problem of relying on human experts for feature selection and improves autonomy. At the same time, the attention mechanism can fill the gap between laboratory results and

industrial practice, which can be an excellent parallel computing feature and improve the speed of model prediction in industrial practice. Finally, the proposed method can be fully embedded into the control and optimization integration framework to provide critical information about the system [8]. Through comparative analysis, the spatio-temporal attention mechanism method is superior to other methods in the evaluation indexes of RMSE and score. Especially in the multi-operation environment, the performance of this method is more stable than other methods. Of course, there are still some problems that need to be further explored. First, the potential harm caused by the situation that the predicted value is greater than the real value is relatively large, but at present, there is no good method to balance the relationship between reducing the predicted value and improving the prediction accuracy; second, there are many types of industrial data, but there may be some potential common features between degraded data. How to extract and use these common features to predict RUL and migrate to different scenarios is also a difficult problem; third, there are uncertainties and inaccuracies in available data, and how to eliminate these problems is also a difficulty in current research. At present, fuzzy logic [27] can generate the best fuzzy rule base in the learning process, which is a solution.

### Data Availability

The data set used in the experiment is the C-MAPSS dataset of NASA. This dataset was generated with the C-MAPSS simulator. C-MAPSS stands for “Commercial Modular Aero-Propulsion System Simulation”, and it is a tool for the simulation of realistic large commercial turbofan engine data. For more information, please visit: <https://github.com/uuuf9/CMAPSSData>.

### Conflicts of Interest

No conflicts of interest exist in the submission of this manuscript, and the manuscript is approved by all authors for publication. The authors would like to declare on behalf of my co-authors that the work described was original research that has not been published previously and is not under consideration for publication elsewhere, in whole or in part. All the authors listed have approved the manuscript that is enclosed.

### Acknowledgments

This work is supported by Natural Science Foundation of China (No. 61871432, No. 61771492), the Natural Science Foundation of Hunan Province (No. 2020JJ4275, No. 2019JJ6008, and No. 2019JJ60054).

### References

- [1] Y. Yu, X. Si, C. Hu, and J. Zhang, “A review of recurrent neural networks: LSTM cells and network architectures,” *Neural Computation*, vol. 31, no. 7, pp. 1235–1270, 2019.
- [2] S. Albawi, T. Abed Mohammed, and S. Al-Zawi, “Understanding of a convolutional neural network,” *Proceedings of the 2017 International Conference on Engineering and Technology (ICET)*, pp. 1-6, IEEE, Antalya, Turkey, August 2017.
- [3] F. O. Heimes, “Recurrent neural networks for remaining useful life estimation,” *Proceedings of the 2008 International Conference on Prognostics and Health Management, USA*, pp. 1-6, Denver, CO, USA, October 2008.
- [4] M. Yuan, Y. Wu, and Li Lin, “Fault diagnosis and remaining useful life estimation of aero engine using LSTM neural network,” in *Proceedings of the 2016 IEEE international conference on aircraft utility systems (AUS)*, IEEE, Beijing, China, October 2016.
- [5] J. Wu, K. Hu, Y. Cheng, H. Zhu, X. Shao, and Y. Wang, “Data-driven remaining useful life prediction via multiple sensor signals and deep long short-term memory neural network,” *ISA Transactions*, vol. 97, pp. 241–250, Feb. 2020.
- [6] Y. Duan, H. Li, M. He, and D. Zhao, “A BiGRU autoencoder remaining useful life prediction scheme with attention mechanism and skip connection,” *IEEE Sensors Journal*, vol. 21, no. 9, pp. 10905–10914, 2021.
- [7] 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.
- [8] Y. Jiang, S. Yin, J. Dong, and O. Kaynak, “A review on soft sensors for monitoring, control, and optimization of industrial processes,” *IEEE Sensors Journal*, vol. 21, no. 11, pp. 12868–12881, 2021.
- [9] L. Wen, Y. Dong, and L. Gao, “A new ensemble residual convolutional neural network for remaining useful life estimation,” *Mathematical Biosciences and Engineering*, vol. 16, no. 2, pp. 862–880, 2019.
- [10] J. Li, X. Li, and D. He, “Domain adaptation remaining useful life prediction method based on AdaBN-DCNN,” *Proceedings of the 2019 Prognostics and System Health Management Conference (PHM-ingdao)*, pp. 1-6, Qingdao, China, October 2019.
- [11] H. Li, W. Zhao, Y. Zhang, and E. Zio, “Remaining useful life prediction using multi-scale deep convolutional neural network,” *Applied Soft Computing*, vol. 89, Article ID 106113, 2020.
- [12] N. G. Borst, *Adaptations for CNN-LSTM Network for Remaining Useful Life Prediction: Adaptable Time Window and Sub-network Training*, Delft University of Technology, Delft, The Netherlands, 2020.
- [13] I. Remadna, S. L. Terrisa, R. Zemouri, S. Ayad, and N. Zerhouni, “Leveraging the power of the combination of CNN and Bi-directional LSTM networks for aircraft engine RUL estimation,” *Proceedings of the 2020 Prognostics and Health Management Conference (PHM-esançon)*, pp. 116-121, Besancon, France, May 2020.
- [14] W. Chen, W. Chen, H. Liu, Y. Wang, C. Bi, and Y. Gu, “A RUL prediction method of small sample equipment based on DCNN-BiLSTM and Domain Adaptation,” *Mathematics*, vol. 10, no. 7, p. 1022, 2022.
- [15] Y. Jiang, Y. Lyu, Y. Wang, and P. Wan, “Fusion network combined with bidirectional LSTM network and Multiscale CNN for remaining useful life estimation,” *Proceedings of the 2020 12th International Conference on Advanced Computational Intelligence (ICACI)*, pp. 620-627, Yunnan, China, March 2020.

- [16] A. Vaswani, *Attention Is All You Need*, Accessed: Dec. 04, 2021. [Online]. Available: <http://arxiv.org/abs/1706.03762>, 2017.
- [17] <https://ww2.mathworks.cn/help/predmaint/ref/prognosability.html>.
- [18] S. Patro and K. Kumar Sahu, "Normalization: a preprocessing stage," arXiv preprint arXiv:1503.06462, 2015.
- [19] A. Saxena, K. Goebel, D. Simon, and N. Eklund, "Damage propagation modeling for aircraft engine run-to-failure simulation," Proceedings of the 2008 International Conference on Prognostics and Health Management, pp. 1-9, Denver, CO, USA, October 2008.
- [20] L. Liu, D. Liu, Y. Zhang, and Y. Peng, "Effective sensor selection and data anomaly detection for condition monitoring of aircraft engines," *Sensors*, vol. 16.5, p. 623, 2016.
- [21] A. Labach, H. Salehinejad, and S. Valaei, "Survey of dropout methods for deep neural networks," arXiv preprint arXiv:1904.13310, 2019.
- [22] E. Phaisangittisagul, "An analysis of the regularization between L2 and dropout in single hidden layer neural network," Proceedings of the 2016 7th International Conference on Intelligent Systems, Modelling and Simulation (ISMS), pp. 174-179, IEEE, Bangkok, Thailand, January 2016.
- [23] A. Listou Ellefsen, E. Bjørlykhaug, V. Æsøy, S. Ushakov, and H. Zhang, "Remaining useful life predictions for turbofan engine degradation using semi-supervised deep architecture," *Reliability Engineering & System Safety*, vol. 183, pp. 240-251, 2019.
- [24] 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.
- [25] S. Zheng, K. Ristovski, A. Farahat, and C. Gupta, "Long short-term memory network for remaining useful life estimation," Proceedings of the 2017 IEEE International Conference on Prognostics and Health Management (ICPHM), pp. 88-95, Dallas, TX, USA, June 2017.
- [26] 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, 2017.
- [27] M. Cacciola, "Aspects about air pollution prediction on urban environment," in *Proceedings of the 4th IMEKO TC19 Symposium on Environmental Instrumentation and Measurements 2013: Protection Environment, Climate Changes and Pollution Control*, pp. 15-20, June 2013.