Research on Aircraft Engine Fault Diagnosis Based on CNN-BiGRU-Attention Model

Chenfeng Jin[1,*], Jie Bai[2]

[1]School of Aeronautical Engineering, Civil Aviation University of China, Tianjin, China, 300300

[2]Tianjin Key Laboratory of Civil Aircraft Airworthiness and Maintenance, Tianjin, China, 300300

*Corresponding author: j_cf123456@163.com

Keywords: Aero Engine Fault Diagnosis, Convolutional Neural Network, Bidirectional Gated Recurrent Unit, Self-Attention Mechanism

Abstract: To enhance the accuracy of fault diagnosis in aircraft engines and improve the ability to capture critical information, this study proposes a model that integrates attention mechanisms with a Convolutional Neural Network and Bidirectional Gated Recurrent Unit (CNN-BiGRU). The Convolutional Neural Network (CNN) processes input data through multiple convolutional and pooling layers, effectively extracting spatial features. The BiGRU helps the model capture contextual dependencies, providing comprehensive dynamic analysis by processing both forward and backward data streams. The self-attention mechanism(Attention) enhances the focus on critical information in fault diagnosis through dynamic weight allocation. The integration of the CNN has significantly improved the model's feature extraction capabilities, likewise, the incorporation of the self-attention mechanism has strengthened its ability to capture essential information. Validated on NASA's C-MAPSS dataset, Compared with the most advanced models, the proposed framework achieves better performance. This model is not only efficient in training but also holds promising prospects for applications in fault prediction within the aviation field.

1. Introduction

The gas path components of an aeroengine may cause material damage and performance degradation if they are operated under extreme conditions for a long time., which may lead to material damage and performance degradation. In such scenarios, issues like excessive tip clearance, turbine blade burning, fuel nozzle clogging, rotor disk or blade fracture can occur [1]. These faults will significantly alter the engine's gas path measurement parameters. Existing models often have low diagnostic accuracy, primarily due to two reasons: space factors: Traditional fault diagnosis models are limited in handling complex spatial features. While some models attempt to improve diagnostic capabilities by increasing the dimension and complexity of data input, this often makes it difficult for the model to capture subtle differences in space features in practical applications.

In recent years, aero-engine fault diagnosis has been widely studied as a hotspot. Existing research can be broadly classified into model-based, data-driven, and hybrid methods. The model method is developed in the form of physical model to evaluate and predict the health status. Im et al. introduced an online fault diagnosis technique using a physical model to estimate fault severity through negative sequence currents [2]. However, these methods face limitations when dealing with complex nonlinear and non-stationary data. With the application of deep learning techniques in fault diagnosis has attracted widespread attention, especially for handling time-series data. For instance, Zuo et al [3]. proposed an aero engine fault diagnosis model, which combines 1DCNN-BiLSTM and CBAM to automatically extract fault features. Kexin Zhang et al. analyzed various sensor data and proposed a deep neural network method for diagnosing aero-engine surge faults, achieving over 99% classification accuracy [4]. Another approach is data-driven, incorporating methods like deep learning networks [5-6]. For example, Wang, Y and Liu, Y take the method combining Convolutional Neural Networks and Long Short-Term Memory (LSTM) networks has been implemented to achieve intelligent fault diagnosis for gas turbine engines [7]. Shen Y et al [8]. proposed a hybrid multi-mode

  DOI: 10.25236/meimie.2024.002

machine learning fault diagnosis strategy for aircraft gas turbine engines.

With the rapid development of computer technology, people have begun to use not only data-driven methods but also artificial intelligence methods based on deep learning. In the field of fault diagnosis, diagnostic methods based on deep learning have also been widely applied and have achieved significant results. For example, on fault datasets of rotating machinery like the Mechanical Fault Prevention Technology Society dataset in the United States, have achieved the highest levels of accuracy. However, fault datasets collected from bearings or gearboxes in stable conditions exhibit cyclostationary characteristics, with single data patterns that are easily overfitted. Therefore, in recent years, experts have proposed numerous models to optimize aero-engine fault diagnosis. Therefore various scholars have used multiple deep learning models, including RNN (Recurrent Neural Network), LSTM, Gated Recurrent Unit (GRU) and Bidirectional Gated Recurrent Unit (BiGRU) models, to address related issues. Among them, the RNN model has shown good performance in fault diagnosis, especially in handling time series data. This model improves the accuracy of fault identification by obtaining the dependency information between data. However, RNNs suffer from gradient vanishing and exploding problems, which degrade their performance when processing long time series data. Additionally, RNNs have limited performance in complex pattern recognition and distinguishing between multiple fault types [9]. Following RNN, the LSTM proposed by Gang Sun and others effectively addresses the gradient vanishing problem of RNNs through its special gating mechanism, improving diagnostic stability and accuracy. However, LSTM-based models have issues with long training times and high computational complexity [10]. Experts then proposed the GRU model to simplify the LSTM network structure, reducing the number of parameters and enhancing computational efficiency. Studies have shown that GRU performs as well as or even better than LSTM in handling time series data, with faster training speed and lower computational resource consumption [11]. Despite its generally good performance, GRU still lags behind LSTM in handling some complex long sequence data and has limitations in distinguishing between various fault modes [12]. Thus, Zheng Cheng and others proposed the BiGRU model, which improves understanding of time series data by leveraging bidirectional information flow. Notably, the 1DCNN-BiLSTM model proposed by Zheng Cheng and colleagues combines the advantages of BiGRU, significantly enhancing fault classification accuracy, and improving the reliability of aircraft engine operations. However, the BiGRU model has high computational complexity and long training times. Additionally, the bidirectional structure may increase latency in some real-time applications.

In summary, to address the issues of high computational complexity, latency, and complex parameter tuning in BiGRU models for fault diagnosis prediction, this study proposes a CNN-BiGRU model based on an attention mechanism. Convolutional Neural Networks (CNN) process input data through multiple convolution and pooling layers, effectively extracting spatial features. The BiGRU upgradation this model's capability to capture temporal dependencies in time series data by analyzing both forward and backward information simultaneously, providing comprehensive dynamic analysis. The self-attention mechanism strengthens the model's focus on critical information in fault diagnosis through dynamic weight allocation. By incorporating CNNs, the model significantly improves feature extraction capabilities.

2. Model and theoretical methods

2.1 CNN-BiGRU-Attention

The structure of the CNN-BiGRU-Attention model is shown in Figure 1. This model combines CNN, BiGRU, and the attention mechanism to improve the performance of fault diagnosis.

Structural details are described below:

1) Input layer: This layer receives time series data from the aero-engine. This data typically includes sensor readings such as temperature, pressure, and speed.

2) CNN layer: Multiple convolutional layers are used for feature extraction. These layers identify local patterns and features in the datSa through convolution operations, which are essential for understanding hidden information in complex engine data. In this model, this paper specify 64 output

filters (convolutional kernels) for the convolutional layers.

3) BiGRU layer: The data is then fed into the BiGRU layer. BiGRU effectively captures temporal dependencies in time series data, and its bidirectional structure allows the model to consider both past and future information simultaneously, which is crucial for time series analysis.

4) Attention mechanism layer: The features output from the previous layer are weighted by the attention mechanism. enabling the model to focus on the most critical information.

5) Others: Finally, the fully connected layers integrate all the learned features and connect them to the last layer. The output layer then makes the final fault diagnosis based on the integrated features.

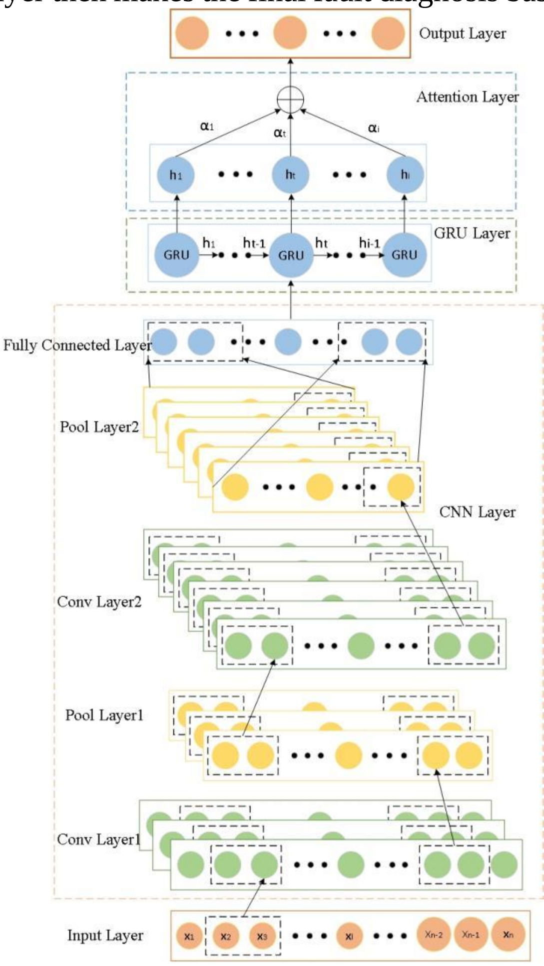

Fig.1 Structure of CNN-BiGRU-Attention

2.2 CNN Method

CNN is a deep learning architecture, primarily consisting of many layers as shown in Figure 2.

In the convolutional layers of CNN, the input data which is convolved with convolutional kernels. These kernels slide over local regions of the input data, generating feature maps through the computation of dot products.

The purpose of the pooling layer is to effectively reduce the feature dimension and thus reduce the computational complexity and stability.

Finally, the output of the pooling layers is flattened and connected into one dimension through one or more fully connected layers for further processing, such as classification or other tasks.

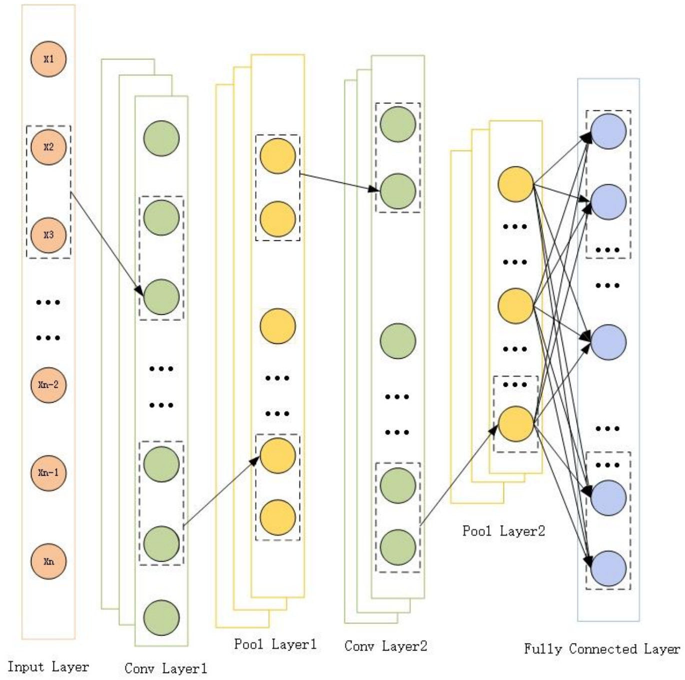

Fig.2 Structure of CNN

2.3 BiGRU

BiGRU is a deep learning architecture, primarily consisting of many layers as shown in Figure 3. Assuming x_t is the input vector at time *t*, the process of the gated recurrent unit is as follows:

$$z_t=\sigma(W_z x_t+U_z h_{t-1}+b_r) \tag{1}$$

$$r_t=\sigma(W_r x_t+U_r h_{t-1}+b_r) \tag{2}$$

$$\tilde{h}_t= tanh(W_h x_t+U_h(r_t h_{t-1})+b_h) \tag{3}$$

$$h_t=z_t h_{t-1}+(1-z_t)\tilde{h}_t \tag{4}$$

where z_t and r_t represent the update gate and reset gate, respectively. $\tilde{h}_t$ represents the candidate hidden state. h_{t-1} and h_t represent the hidden states at time t-1 and t respectively. W and U represent weights. *b* represent the bias term; and σ represents the Sigmoid function. The mathematical expressions for the BiGRU network structure are as follows:

$$\overrightarrow{h}_t=GRU\,f(x_t,\overrightarrow{h}_{t-1}) \tag{5}$$

$$\overleftarrow{h}_t=GRU\,f(x_t,\overleftarrow{h}_{t-1}) \tag{6}$$

$$h_t=f(W_{\overrightarrow{h_t}}\overrightarrow{h}_t+W_{\overleftarrow{h_t}}\overleftarrow{h}_t+b_t) \tag{7}$$

where $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$ represent the forward and backward hidden states at time t respectively. $W_{\overleftarrow{h_t}}$ and $W_{\overrightarrow{h_t}}$ represent the weights of the forward and backward hidden states at time *t* respectively; and b_t is the bias term for the hidden state at time *t*.

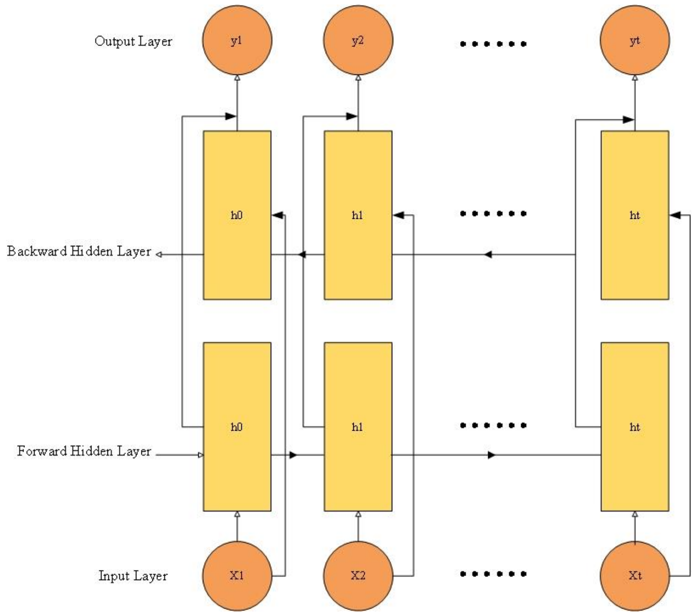

Fig.3 Structure of BiGRU

2.4 SE-Attention

The model structure diagram shown in Figure 4 illustrates the detailed computation process and principles of the Attention mechanism:

Phase One: Calculating Attention Scores

In this phase, the model needs to calculate the similarity between the Query and each Key. This is usually achieved through dot product, cosine similarity, or a multi-layer perceptron (MLP). In the first phase, each Key undergoes a dot product operation with the Query to obtain an initial similarity score. These scores represent the degree of match between the Query and each Key, indicating the importance of each Key to the Query.

The formula is expressed as follows:

$$Similarity\left(Query,\ Key_i\right)=Query \cdot Key_i \tag{8}$$

$$Similarity\left(Query,\ Key_i\right)=\frac{Query \cdot Key_i}{\|Query\| \cdot \|Key_i\|} \tag{9}$$

$$Similarity\left(Query,\ Key_i\right)=MLP\left(Query,\ Key_i\right) \tag{10}$$

Along the formula, Query is the query vector, and $\mathrm{Key}_{\mathrm{i}}$ is the i-th key vector.

Phase Two: Softmax Normalization

In the second phase, the obtained attention scores are scaled (usually by dividing by the square root of the dimensionality of the Key vector) to avoid gradient vanishing or explosion problems. Then, they are normalized using the Softmax function. Softmax ensures that the sum of all output scores equals 1, allowing each score to be interpreted as a probability. This step enables the model to emphasize the most relevant features and suppress irrelevant information.

The formula is expressed as follows:

$$a_i=Softmax(Sim_i)=\frac{e^{Sim_i}}{\sum_{j=1}^{L_x} e^{Sim_j}} \tag{11}$$

where L_x is the sequence length, as the total number of keys, and Sim_i represents the dot product between the query vector and the i-th key vector.

Phase Three: Calculating Weighted Value

In the final phase, each Value is weighted according to the normalized scores obtained from the Softmax step. The weighted sum ultimately constitutes the output Attention Value, which reflects the aggregated information of all Values weighted by their importance according to the Query.

The formula is expressed as follows:

$$Attention(Query, Source) = \sum_{i=1}^{L_x} a_i \ Value_i \tag{12}$$

where $Value_i$ is the value vector, and a_i represents the weight of i-th vector, calculated as the corresponding Softmax score through cumulative summation.

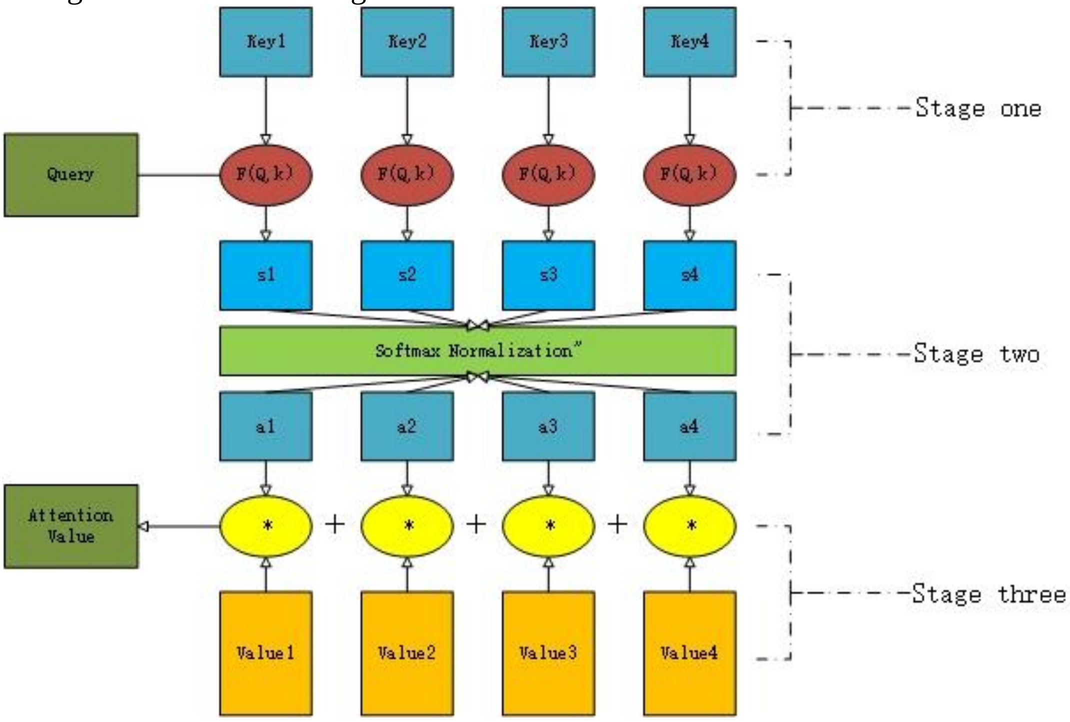

Fig.4 Structure of Attention

3. Experiments and analysis

3.1 Simulation setup

(1) Experimental environment

Table.1. Experimental environment table

Software and hardware configuration	Attributes
Operating system	64-bit Windows 10 operating system
GPU	Intel Xeon E5-2650 v4 Processor
GPU memory	6.0 GB
Graphics card	NVIDIA GeForce GTX 1660 Ti
Programming language	Python3.8
Framework	Tensorflow

The experimental environment configuration is shown in Table 1. This paper used the TensorFlow framework to construct and train deep learning models. All models were written in the Python 3.8 programming language. Additionally, the Keras library integrated into TensorFlow provides API modules for existing deep learning network models. By calling these APIs, network models can be easily constructed and parameter tuning is very convenient.

(2) Data

The dataset used in this paper is the engine full-life simulation experiment dataset (C-MAPSS) from the National Aeronautics and Space Administration (NASA). Which was designated for research on aero-engine fault prediction at the International Conference on Prognostics and Health Management (PHM08). The C-MAPSS dataset records the full-cycle sensor monitoring data of a certain type of turbofan engine under different fault modes and operating conditions, from initial operation to gradual failure. The dataset includes three operational setting parameters, 21 sensor status monitoring parameters, and the corresponding number of operational cycles.

The C-MAPSS dataset is a simulated dataset generated by NASA using the Commercial Modular Aero-Propulsion System Simulation software. The dataset covers key components of aero-engines, such as fans, compressors, combustion chambers, turbines, and exhaust nozzles, demonstrating extensive application value. It can be used not only to compare and evaluate the performance of different engine models but also to conduct deep analysis of the working characteristics of each component and their specific impacts on overall engine performance. There are four subsets in total, with different numbers of operating conditions and fault states, as shown in Table 2.

Table.2. C-MAPSS dataset

Dataset	FD001	FD002	FD003	FD004
Training Set	100	260	100	249
Test Set	100	259	100	248
Operating Conditions	1	6	1	6
Fault States	1	1	2	2

In this study, this model focus on the FD001 dataset under single operating conditions, which involves dividing the FD001 dataset into training and testing subsets. The training subset, Train_FD001.txt, includes parameter information for 100 engines operating throughout their entire lifecycle. The testing subset, Test_FD001.txt, includes parameter information for 100 engines that stopped at a certain point before failure. This information mainly consists of data from multiple sensors. Each engine's parameter information includes three flight condition monitoring parameters (altitude, Mach number, throttle resolver angle) and 21 performance monitoring parameters, totaling 24 sensor monitoring parameters. This paper primarily studies the sensor parameter information of an engine throughout its lifecycle.

Due to different operating conditions and fault modes, the time points recorded during single conditions, i.e., during aircraft cruising, can be approximately considered constant for the operating condition parameters (altitude, Mach number, and throttle resolver angle). Randomly selecting six columns of data from FD001, namely LPC outlet total temperature, HPC outlet total temperature, LPT outlet total temperature, HPC outlet pressure, fan speed, and core engine speed, the sensor characteristic data for engine 1 throughout its lifecycle is shown in Table 3:

Table.3. Partial sensor data

Serial Number	Parameter	Meaning
7	T24	Total temperature at the low-pressure compressor outlet
8	T30	Total temperature at the high-pressure compressor outlet
9	T50	Total temperature at the low-pressure turbine outlet
12	P30	Total pressure at the high-pressure compressor outlet
13	Nf	Fan speed
14	Nc	Core speed

(3) Parameter settings and processing flow

Table 4 records the initial learning rate, batch size, and optimizer for the CNN-BiGRU-Attention model details.

Table.4. CNN-BiGRU-Attention model training parameters

Training parameter	Value
Learning rate	0.001
Number of iterations	10
Convplution kernel size	1
Dropout rate	0.3
Number of convolution kernels	64
Number of GRU layer units	64
Input dimension	6
Optimizer	Adam

(4) Evaluation metrics

In the field of machine learning, a confusion matrix is generally used to evaluate classification results in supervised learning. In the unsupervised learning, this matrix is usually called a matching matrix.

Accuracy is a metric that indicates the proportion of correctly predicted samples out of the total number of samples, defined as follows:

$$Acc=\frac{TP+TN}{TP+TN+FP+FN} \tag{13}$$

Precision is defined as the proportion of actual positive samples among the predicted positive samples, defined as follows:

$$Pre=\frac{TP}{TP+FP} \tag{14}$$

Recall is defined as the proportion of correctly identified positive samples among all actual positive samples, defined as follows:

$$Rec=\frac{TP}{TP+FN} \tag{15}$$

The F1-Score is a metric that comprehensively reflects the precision and recall of a classification model, particularly effective in handling imbalanced datasets, defined as follows:

$$F1=2*\frac{Precision.Recall}{Precision+Recall} \tag{16}$$

3.2 Model validation

To further evaluate the performance of the proposed model on the training set and test set, and analyze the model's generalization ability and fitting condition, this model compared the loss values (loss) and validation loss values (val_loss) on the dataset. The comparison results are shown in Figure 5.

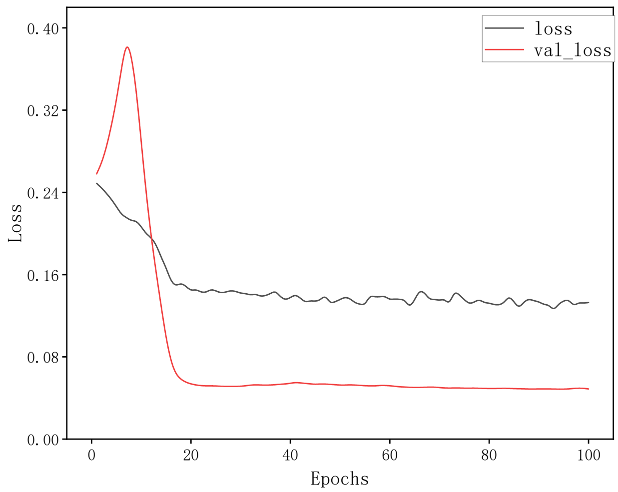

Fig.5 Comparison chart of loss values on training and validation sets for the model

Based on Figure 5, this model can observe the changes in loss values during the training process. The val_loss initially shows significant fluctuations with an upward trend, indicating poor adaptation of the model to the test data in the early stages. Subsequently, val_loss rapidly decreases and stabilizes, indicating that the model parameters begin to adapt and learn effective features. After approximately 20 epochs, both the val_loss and loss stabilize, and val_loss remains at a relatively low level, suggesting that the CNN-BiGRU-Attention model demonstrates good adaptability and generalization capability on both the training and validation datasets. The loss value is consistently lower than the val_loss, with a relatively small gap, indicating that the test set exhibits larger errors, suggesting a slight overfitting phenomenon. Overall, the model's performance remains reasonable.

3.3 Ablation experiments

Ablation experiments (horizontal comparison experiments) involve gradually adding or removing model components to evaluate the specific impact of these components on model performance. For the CNN-BiGRU-Attention model in this study, the contributions of each component are analyzed following.

As the Table 5 below, the evaluattion indicators of the above components are summarized in Table 5.

Table.5. Evaluation metrics for different models on the C-MAPSS dataset

Model name	Precision	Recall	Accuracy	F1 Score
BiGRU	0.8219	**0.8824**	0.8772	**0.8511**
CNN- BiGRU	0.8789	0.8088	0.8830	0.8461
CNN-BiGRU-Attention	**0.8836**	0.8031	**0.8965**	0.8414

According to the results in Table 5, although the inclusion of the CNN layer reduces the recall rate, it enhances the precision and accuracy of the model's local feature extraction capabilities. Specifically, the precision increases by 0.057 and the accuracy improves by 0.0058. Moreover, after incorporating the Attention mechanism, the model's accuracy and precision further improve compared to the previous models. Specifically, the precision increases by 0.0135 and the accuracy increases, indicating that the Attention layer plays a crucial role in enhancing the model's focus on important time steps. These findings demonstrate the importance of the CNN and Attention layers in improving the predictive performance of the model, especially when dealing with complex time-series data. The proposed model shows the best performance in terms of accuracy, but the lowest F1 score, indicating a relatively low recall rate. The Attention mechanism may help the model concentrate on learning specific features, leading to the neglect of some positive samples. The CNN-BiGRU-Attention model fails to identify all true positive samples, resulting in some positive samples being missed.

The dataset used in this study is the engine full-life simulation dataset (C-MAPSS) provided by NASA. Specifically, our experiments focus on the FD001 dataset under a single operating condition, which includes sensor readings and fault modes of aircraft engines under different operating cycles. The complexity of these data is mainly reflected in the non-linearity and non-stationarity of the time series, making it difficult for traditional machine learning methods to capture potential dynamic patterns and trends. By using the CNN-BiGRU-Attention model, this model can more effectively identify and utilize the critical information in these complex time-series data, thereby improving the accuracy of sensor prediction data.

To comprehensively analyze the effectiveness of the proposed method, the loss comparisons of these three models are shown in Figure 6:

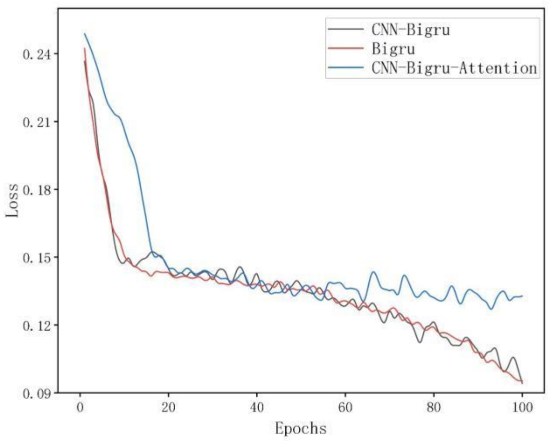

Fig.6 Loss comparison of BiGRU and its variants

Based on Figure 6, this paper can easily observe that the loss value of the proposed model is relatively high due to the excessive number of parameters in the early training stage, leading to overfitting. Additionally, although the Attention mechanism brings the advantage of focusing on critical features, it fails to function effectively in the early training stage, resulting in higher loss values in the initial iterations. Among them, CNN-BiGRU-Attention shows an upward trend in the number of iterations from 60 to 80, overfitting during the training process. Finally, the loss values of all three curves stabilize after 100 iterations.

3.4 Superiority verification

Based on the model training parameters set forth above, the experiments in this article will be conducted on the C-MAPSS Train_FD001.txt dataset. The predicted values obtained from this model are compared with the test set divided from Train_FD001.txt, and their performance is evaluated using four evaluation metrics: precision, recall, accuracy and F1 score. Compared with other algorithms, the evaluation results are shown in Table 6:

Table.6. Evaluation metrics for different models on the C-MAPSS dataset

Model name	Precision	Recall	Accuracy	F1 Score
CNN-Bigru-Attention	0.8836	0.8031	0.8965	0.8414
CNN-Bilstm-Attention	0.8059	0.7941	0.8421	0.7969
CNN-Attention-Bilstm	0.8002	0.7647	0.8304	0.7819
CNN-Attention-Bigru	0.7764	0.7794	0.824	0.7794

CNN-Bigru-Attention performs significantly better in terms of precision compared to other models, outperforming the CNN-Attention-Bigru model by approximately 13.82%, the CNN-Attention-Bilstm model by 10.42%, and the CNN-BiLSTM-Attention model by 9.64%. This indicates that CNN-Bigru-Attention is more accurate in identifying positive samples with a lower false positive rate. In terms of accuracy, CNN-Bigru-Attention significantly outperforms other models, with an improvement of approximately 8.80% over the CNN-Attention-Bigru model, 6.46% over the CNN-Bilstm-Attention model, and 7.95% over the CNN-Attention-Bilstm model. This shows that CNN-Bigru-Attention has a higher overall correct prediction rate. In terms of recall, although the overall improvement is small, CNN-Bigru-Attention still outperforms other models. Specifically, it improves by about 3.04% compared to the CNN-Attention-Bigru model, 1.13% compared to the CNN-Bilstm-Attention model, and 5.01% compared to the CNN-Attention-Bilstm model. This indicates that CNN-Bigru-Attention performs better in capturing all positive samples. To comprehensively consider precision and recall, CNN-Bigru-Attention's F1 score is also significantly higher than other models, with an improvement of approximately 7.95% over the CNN-Attention-Bigru model, 7.61% over the CNN-Attention-Bilstm model, and 5.59% over the CNN-Bilstm-Attention model. This shows that CNN-Bigru-Attention performs well in balancing precision and recall.

Overall, these advantages demonstrate that CNN-Bigru-Attention has higher reliability and practicality in the actual application of the C-MAPSS dataset. The superiority of CNN-Bigru-Attention has been verified.

4. Conclusion

Traditional condition monitoring methods often rely on threshold determination, which not only makes it difficult to capture the subtle temporal changes of engine component faults but also fails to effectively predict potential faults. Therefore, this paper introduces a CNN-BiGRU-Attention-based model. The model first utilizes the powerful feature extraction capabilities of the CNN layer to identify local patterns and key features in sensor data, which is crucial for understanding the operating state of the engine. Then, the BiGRU layer, through its bidirectional structure, integrates information from both forward and backward time steps, effectively capturing the dynamic characteristics of time-series data. Additionally, the Attention mechanism allows the model to focus on critical time steps, optimizing information processing and enhancing the analysis of complex data. Using the Adam optimizer significantly improves training speed, stability, and prediction performance through efficient parameter optimization.

Applying the CNN-BiGRU-Attention model to the C-MAPSS engine simulation dataset shows good classification results, with an accuracy of 0.8965 and a precision of 0.8836. The model's effectiveness and applicability have been verified.

In summary, the model demonstrates excellent performance in fault diagnosis due to its high precision and accuracy, fulfilling the stringent demands of aircraft engine fault diagnosis and guaranteeing system safety and reliability. The paper’s model has a momentous potential for improving aircraft engine fault diagnosis. It can be applied in predictive maintenance, real-time monitoring, and comprehensive engine health management systems. By accurately predicting faults and analyzing engine data, it enhances maintenance scheduling, reduces downtime, and ensures operational safety. Future applications include expanding the model to other engine types, integrating with IoT and big data, and applying it across different industries like automotive and power generation. Additionally, developing user-friendly diagnostic tools and aligning with regulatory standards will facilitate broader adoption. The model's adaptability and high accuracy promise substantial advancements in fault diagnosis technology.

References

[1] Xia C J, Zhan Y Y. Research status of gas path fault diagnosis for aeroengine based on data-driven[J]. Scinece Technology and Engineering, 2022, 22(5): 1741-1750.

[2] Fentaye A D, Zaccaria V, Kyprianidis K. Aircraft engine performance monitoring and diagnostics based on deep convolutional neural networks[J]. Machines, 2021, 9: 337.

[3] Wu J, Kong L, Kang S, Zuo H, et al. Aircraft engine fault diagnosis model based on 1DCNN-BiLSTM with CBAM[J]. Sensors, 2024, 24: 80.

[4] Zhang K, Lin B, Chen J, Wu X, et al. Aero-engine surge fault diagnosis using deep neural network [J]. Computer Systems Science and Engineering, 2022, 42(1): 351-360.

[5] Zhou H W, Huang J Q, Lu F. Echo state kernel recursive least squ ares algorithm for machine condition prediction[J]. Mechanical Systems & Signal Processing, 2018, 111: 68-86.

[6] Cui J G, Li G Q, Jiang L Y, et al. Fault diagnosis of aircraft engines based on deep self coding networks [J]. Vibration, Testing and Diagnosis, 2021, 41 (1): 8589

[7] Wang Y, Liu Y, Zhang Y. Intelligent fault diagnosis of gas turbine engine using convolutional neural network and long short-term memory [J]. Applied energy, 2021, 143(4): 56-71.

[8] Shen Y, Khorasani K. Hybrid multi-mode machine learning-based fault diagnosis strategies with application to aircraft gas turbine engines [J]. Neural Networks, 2020, 130: 126-142.

[9] Kibrete F, Woldemichael D E. Applications of artificial intelligence for fault diagnosis of rotating machines: A review[C]. Artificial Intelligence and Digitalization for Sustainable Development, vol 455. Springer, Cham, 2022.

[10] Maurya M, Panigrahi I, Dash D, et al. Intelligent fault diagnostic system for rotating machinery based on ioT with cloud computing and artificial intelligence techniques: a review[J]. Soft Comput, 2024, 28: 477–494.

[11] Iqbal M, Madan A K. Artificial intelligence-based bearing fault diagnosis of rotating machine to improve the safety of power system[C]. Lecture Notes in Electrical Engineering, vol 1086. Springer, Singapore, 2023.

[12] Gawde S, Patil S, Kumar S, et al. A scoping review on multi-fault diagnosis of industrial rotating machines using multi-sensor data fusion[J]. Artif Intell Rev, 2023, 56: 4711–4764.