International Journal of Advanced Engineering and Technology Research
ISSN: 3106-8677 (Print), ISSN: 3106-8685 (Online) | Vol. 3, No. 1, 2026

A Multi-Branch Parallel-Fusion Method for Fault Diagnosis of Shearer Bearings

Lei Wang [1], Yingying Han [1, *], Junyan Qi [2], Wen Wang [3]

[1] School of Computer Science and Technology, Henan Polytechnic University, Jiaozuo, China

[2] School of Software, Henan Polytechnic University, Jiaozuo, China

[3] School of Energy Science and Engineering, Henan Polytechnic University, Jiaozuo, China

[*] **Corresponding author**: Yingying Han

Abstract: Conventional single-network models struggle to simultaneously capture local transient impacts, temporal dependencies, and contextual correlations from rolling bearing vibration signals. Moreover, traditional serial hybrid architectures suffer from feature dilution, wherein weak fault features extracted by upstream layers are progressively weakened during forward propagation. To address these limitations, this paper proposes CLTrans, a parallel multi-branch fault diagnosis model integrating a one-dimensional convolutional neural network (1D CNN), a two-layer long short-term memory (LSTM) network, and an enhanced Transformer to extract local, temporal, and global features in parallel. The Transformer branch incorporates Rotary Position Embedding (RoPE) to enhance relative positional awareness of periodic impact patterns, and employs Local Window Attention (LWA) to restrict self-attention computation within overlapping sliding windows, thereby reducing computational complexity from quadratic to linear order while suppressing background noise interference. Features from all three branches are concatenated into a unified joint representation for fault classification. Experiments on the Case Western Reserve University (CWRU) dataset for 10-category bearing fault classification yield a macro-averaged accuracy of 99.57%, precision of 99.55%, recall of 99.29%, and F1-score of 99.40%. Compared with the best-performing baseline, Transformer-BiLSTM, CLTrans improves accuracy by 1.29 percentage points and F1-score by 1.76 percentage points. Ablation studies confirm that the three branches form complementary feature representations. The proposed model advances multi-class fault recognition performance on public bearing datasets and provides a methodological foundation for condition monitoring of shearer cutter bearings under noisy industrial environments.

Keywords: Coal-cutting machine bearings, fault diagnosis, parallel fusion model, Rotary Position Embedding, local window attention.

1. Introduction

Rolling bearings are critical supporting components in rotating machinery, and their health condition directly affects equipment reliability and operational safety [1]. In actual underground fully mechanized mining operations, coal cutters are constantly exposed to extremely harsh industrial environments, which not only causes bearing vibration signals to be accompanied by extremely strong non-stationary random noise but also makes early-stage failure features faint and highly susceptible to being masked by background noise. Therefore, the key to achieving intelligent reliability diagnosis of Shearer Cutting-Unit Bearings lies in how to deeply mine multidimensional, deep-level features from long-time-series vibration signals.

With the development of artificial intelligence, deep-learning-based methods have become an important research direction in rotating machinery fault diagnosis because of their ability to learn fault representations directly from monitoring data [2, 3]. These methods can extract deep features from raw data through cascaded multi-layer nonlinear modules in deep learning models without the need for manual feature extraction, thereby expanding the model's range of applications. For example, Wang et al. [4] proposed a feature-level attention-guided multitask CNN for simultaneous bearing fault diagnosis and operating-condition identification; Fu et al. [5] developed a parallel CNN–LSTM architecture, in which the CNN extracts local features from CWT-based time–frequency representations, while the LSTM models temporal correlations in one-dimensional vibration signals. For instance, CNNs tend to overlook long-term temporal evolution patterns, LSTMs lack sufficient sensitivity to local impact features, and both struggle to efficiently capture global feature correlations. Consequently, researchers both domestically and internationally have conducted extensive studies on hybrid architecture integration and attention mechanisms to address the performance shortcomings of single models. For instance, Sun and Zhao [6] developed an end-to-end 1DCNN–LSTM model for bearing fault diagnosis. Guo et al. [7] combined an attention mechanism with CNN and BiLSTM to enhance bearing fault feature representation. Hao et al. [8] proposed a one-dimensional convolutional LSTM network to jointly extract and fuse spatial and temporal features from multisensor vibration signals. Xu et al. [9] developed a hybrid deep-learning model to improve rolling-bearing fault diagnosis by integrating complementary feature-learning modules. Ding et al. [10] proposed a time–frequency Transformer to extract discriminative representations from vibration time–frequency maps. Hou et al. [11] subsequently developed Diagnosisformer, an improved Transformer architecture for efficient rolling-bearing fault diagnosis.

Overall, the aforementioned studies predominantly adopt traditional unidirectional serial cascading architectures. However, the serial cascading mechanism involves unidirectional linear information flow. The cutting vibration

signals of coal cutters exhibit both the spatial locality of high-frequency transient impacts and the long-term temporal nature of damage evolution. Weak features extracted by the upstream network are highly susceptible to feature dilution and information loss as the number of network layers increases, making it difficult to simultaneously account for and fully preserve complementary features across different dimensions.

To address the feature loss in the aforementioned network architecture, as well as the issues of high noise-induced failure and computational redundancy in traditional Transformers, this paper proposes a fault diagnosis model based on the parallel fusion of CNN, LSTM and an improved Transformer (hereinafter referred to as CLTrans). The core improvements of this paper are as follows: First, a three-branch parallel fusion architecture—local, temporal, and global—is constructed. Abandoning the traditional unidirectional serial cascading, the CNN branch adaptively captures the spatially localized features of transient impacts, the LSTM branch simultaneously models the long-term temporal evolution of faults, and the improved Transformer branch extracts contextual correlation features through overlapping local-window attention. The three feature streams are ultimately concatenated at the feature level and fused at the feature layer, achieving efficient complementarity among multidimensional features and reducing the loss of upstream features caused by serial architectures. On the other hand, the Transformer encoder undergoes in-depth component optimization to address highly noisy industrial operating conditions. Rotary Position Embedding (RoPE) is introduced at the input to leverage the rotational properties of complex space, endowing the time-series signals with innate relative position awareness and enhancing the detection of relative damage phases within the fault impact cycle; simultaneously, a Local Window Attention (LWA) mechanism is introduced, strictly limiting self-attention computations to overlapping sliding windows that cover typical bearing impact cycles. This not only reduces the computational complexity of long sequences to a linear order but also effectively suppresses interference from long-duration steady-state background noise underground through a local focusing mechanism, thereby achieving the "amplification and extraction" of weak fault pulses.

2. Theoretical Knowledge

2.1. Convolutional Neural Networks

Convolutional neural networks (CNN) employ local receptive fields and shared convolutional kernels, and can extract local structural information from input data through convolution kernels [12]. For one-dimensional vibration signals, a one-dimensional CNN performs convolution operations along the time axis, enabling the extraction of local waveform variations and transient impact features. Its basic computational form is as follows:

$$y_i = f\left(\sum_{j=1}^{n} W_{ij} * x_j + b_i\right) \tag{1}$$

Where x_j is the jth input feature; W_{ij} is the parameter of the convolutional kernel connecting the jth input to the ith output; b_i represents the bias of the ith output channel; $*$ denotes the convolution operation; $f(\cdot)$ denotes the nonlinear activation function. Pooling layers can further reduce the feature length, decrease computational complexity, and enhance the stability of local features.

2.2. Long Short-Term Memory Neural Networks

The Long Short-Term Memory (LSTM) network was introduced by Hochreiter and Schmidhuber to model long-term dependencies and mitigate the vanishing-gradient problem in recurrent neural networks [13]. It regulates information flow through forget, input, and output gates, making it suitable for temporal dependencies modeling in vibration sequences. Its structure is shown in Figure 1.

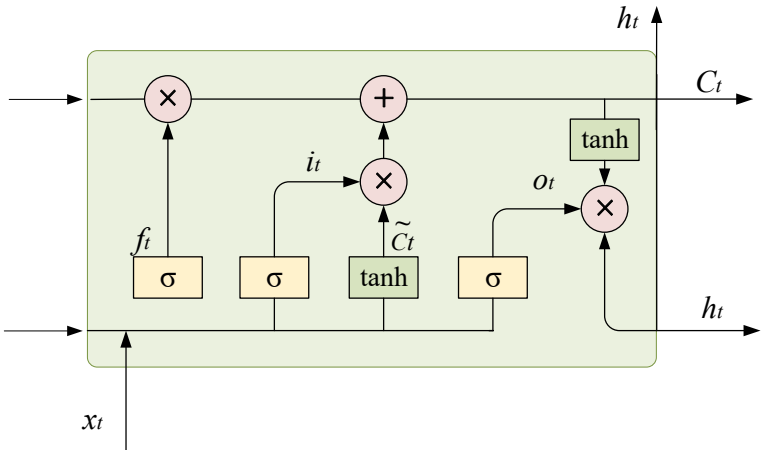


Figure 1. Structure of LSTM network

The state update process is as follows:

$$\begin{cases} f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \\ i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \\ C_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \\ C_t = f_t \odot C_{t-1} + i_t \odot C_t \\ o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \\ h_t = o_t \odot \tanh(C_t) \end{cases} \quad (2)$$

Where x_t is the input at time t; h_{t-1} and C_{t-1} are the hidden state and cell state at the previous time step, respectively; f_t, i_t,and o_t, are the forget gate, input gate, and output gate, respectively; $\tilde{C}$ is the candidate cell state; $\sigma(\cdot)$ is the sigmoid function; $\odot$ denotes the Hadamard product. LSTM uses hidden states to propagate information across different time steps and is suitable for modeling temporal dependencies in vibration sequences.

2.3. Transformer Encoder

The Transformer encoder is a sequence modeling architecture based on the self-attention mechanism, capable of capturing global dependencies between any positions within a sequence. In the task of bearing fault diagnosis, this architecture uses the self-attention mechanism to deeply explore the global dependencies in single-channel vibration signals, accurately capturing the long-range correlation features inherent in the fault evolution process. The network architecture of the Transformer encoder is shown in Figure 2.

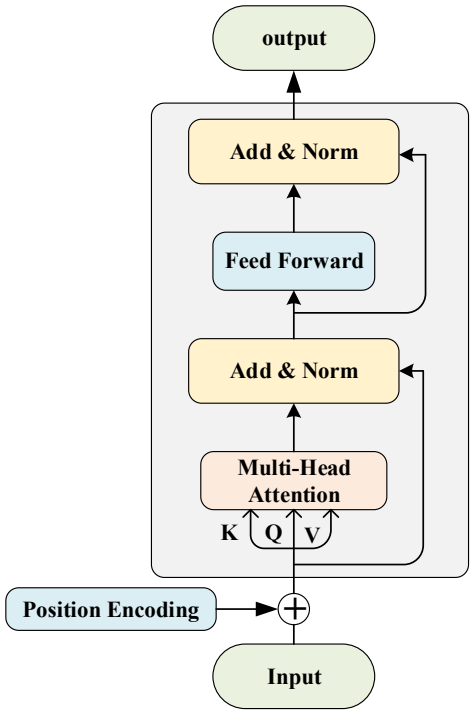


Figure 2. Structure diagram of Transformer encoder

The Transformer encoder primarily consists of multi-head self-attention, a feedforward neural network, residual connections, and layer normalization. The scaled dot-product attention is computed as follows:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (3)$$

Where Q, K, and V represent the query matrix, key matrix, and value matrix, respectively; d_k is the dimension of the key vector; $\sqrt{d_k}$ is a scaling factor used to prevent the dot product values from becoming excessively large as the feature dimension increases, mitigate excessive concentration in the Softmax output, and improve training stability. Multi-head attention maps the input to multiple feature subspaces, and its expression is as follows:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \cdots, \text{head}_n)W^O \quad (4)$$

The attention output undergoes a nonlinear mapping through a feedforward neural network:

$$\text{FFN}(x) = \text{ReLU}(xW_1 + b_1)W_2 + b_2 \quad (5)$$

Where x represents the output of the previous layer; W_1 and W_2 are weight matrices; and b_1 and b_2 are bias terms. The position-wise feed-forward network independently applies the same nonlinear transformation to each sequence position.

Residual connections and layer normalization are used to preserve input information and improve the training stability of deep networks.

3. CLTrans Parallel FUSION Fault Diagnosis Model

This paper proposes CLTrans, a parallel fusion model combining CNN, LSTM, and an improved Transformer, whose structure is shown in Figure 3. The model consists of three independent feature extraction branches, which extract local transient features, temporal dependencies features, and contextual correlation features from the vibration signal, respectively. The outputs from each branch are concatenated to form a joint representation, and fault classification is performed via a fully connected layer and a Softmax function.

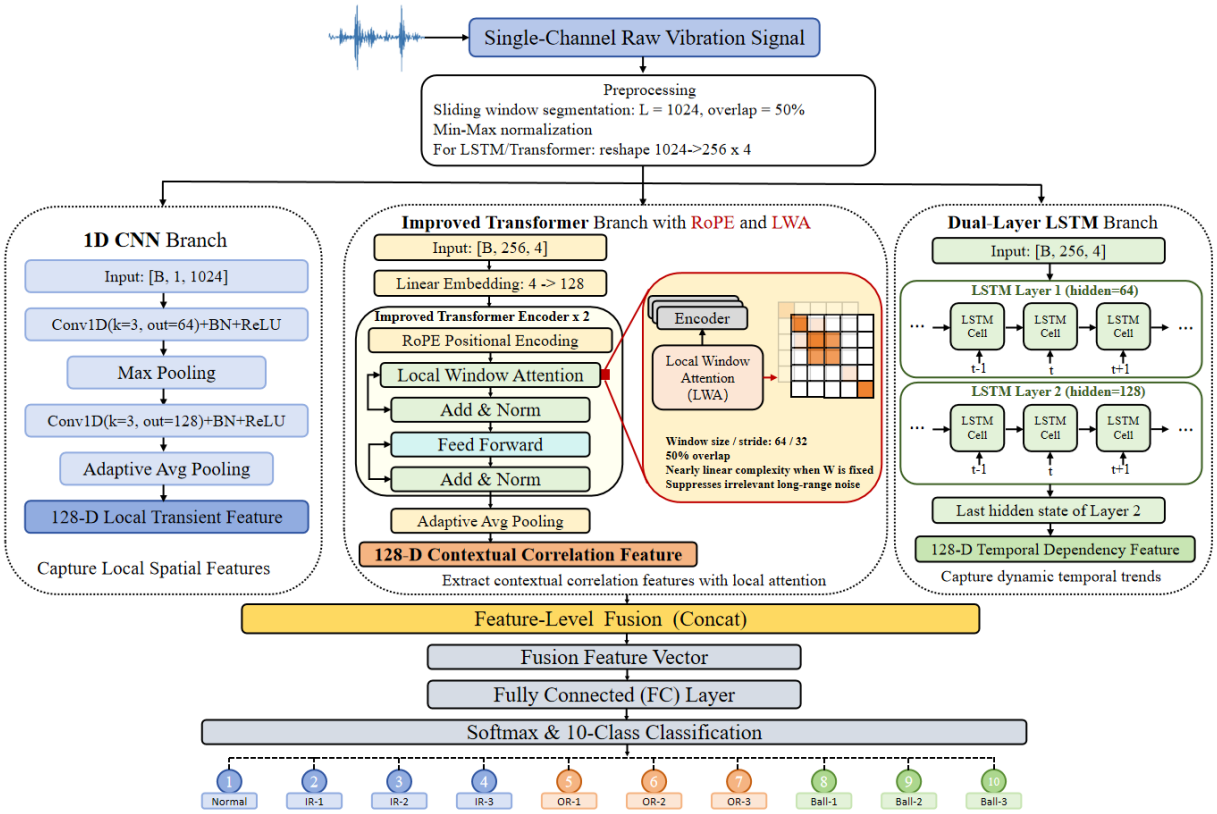


Figure 3. CLTrans parallel-fusion fault diagnosis model

3.1. Design of an Improved Transformer Encoder

When processing long sequences of vibration signals from rolling bearings, traditional Transformers face two major issues: computational complexity that increases quadratically with sequence length, and poor ability to detect periodic impact features using absolute position embeddings. To address this, this paper introduces Rotary Position Embedding (RoPE) and Local Window Attention (LWA) mechanisms to enhance the modeling of relative position information and reduce the computational overhead of attention for long sequences. The specific design is as follows:

(1) Rotary Position Embedding (RoPE): This method uses a rotation matrix to transform absolute position information into relative positional relationships between the query and the key [14]. Its core calculation formula is:

$$\begin{cases} \tilde{q}_m = R_m q_m \\ \tilde{k}_n = R_n k_n \end{cases} \tag{6}$$

Where q_m and k_n are the unposition-encoded query and key vectors at positions m and n, respectively; R_m and R_n are the rotation matrices for the corresponding positions.

The encoded dot product satisfies:

$$q_m^T \tilde{k}_n = q_m^{\mathrm{T}} R_m^{\mathrm{T}} R_n k_n \tag{7}$$

Therefore, the attention scores can incorporate relative positional information between positions m and n, which helps characterize the positional relationships between different time units in the vibration sequence.

(2) Local Window Attention (LWA): Inspired by sliding-window attention for efficient long seque- nce modeling [15], the proposed LWA divides the input vibration sequence into several overlapping local windows and computes self-attention independently within each window. Let the input feature sequence be:

$$H = [h_1, h_2, \ldots, h_L]^{\mathrm{T}} \in \mathbb{R}^{L \times d} \tag{8}$$

Where L is the sequence length, and d is the feature dimension. Let W be the window size and S be the sliding step size; then the number of windows is:

$$N_w = \left[\frac{L - W}{S} \right] + 1 \tag{9}$$

The rth window is represented as:

$$H^{(r)} = [h_{p_r}, h_{p_r+1}, \ldots, h_{p_r+W-1}]^T, \quad p_r = (r-1)S + 1 \tag{10}$$

Where p_r is the starting position of the rth window. A linear mapping is performed on each local window to obtain the query matrix, key matrix, and value matrix $Q^{(r)}$, $K^{(r)}$, and $V^{(r)}$, respectively. The attention calculation within the window is given by:

$$Z^{(r)} = \mathrm{Softmax}\left(\frac{Q^{(r)} K^{(r)T}}{\sqrt{d_k}} \right) V^{(r)}, \quad r = 1,2,\ldots,N_w \tag{11}$$

Where d_k is the dimension of the key vector, and $Z^{(r)}$ is the attention output for the rth window. For temporal positions that appear simultaneously in multiple overlapping windows, the corresponding outputs are averaged to restore a feature representation of the same length as the input sequence.

The computational complexity of attention for a single window is $O(W^2 d)$, so the complexity for all windows is $O(N_w W^2 d)$. When the sliding stride S and window size W maintain a fixed ratio, the complexity can be approximated as $O(LWd)$. In this paper, we set $W = 64$, $S = 32$, and the overlap rate between adjacent windows to 50%. Overlapping windows reduce information truncation caused by local

impacts occurring at window boundaries, while also limiting the contribution of irrelevant information from distant areas to feature aggregation.

3.2. Key Steps of the Model

The CLTrans model performs fault diagnosis through a five-step process: "signal preprocessing—three-branch parallel feature extraction—multidimensional feature fusion—fault classification." The specific steps are as follows:

(1) Signal preprocessing:

The raw vibration signal is divided into samples using a sliding window technique, with each sample set to a length of $L_s = 1024$ and a window overlap of 50%, followed by Min-Max linear normalization. The preprocessed input tensor is $X \in \mathbb{R}^{B\times 1024\times 1}$ (where B is the batch size). For the CNN branch, the input is reshaped into a channel-first format: $X_{CNN} \in \mathbb{R}^{B\times 1\times 1024}$; For the LSTM and Transformer branches, every 4 consecutive sample points are grouped into a single local time unit, resulting in a temporal feature matrix $X_{seq} \in \mathbb{R}^{B\times 256\times 4}$, This operation preserves the original number of sample points and their temporal order but reduces the number of time steps the network needs to process from 1024 to 256, thereby reducing the temporal computational overhead in the model's front end.

(2) Local Transient Feature Extraction:

X_{CNN} is fed into a one-dimensional CNN branch consisting of two convolutional blocks with 64 and 128 output channels, respectively. Each convolutional block comprises a one-dimensional convolution, batch normalization 4, and a ReLU activation function. The first convolutional block is followed by a max-pooling layer to reduce the temporal resolution and computational cost. After the second convolutional block, adaptive average pooling is applied to aggregate the feature maps into a 128-dimensional local transient feature vector $F_{CNN} \in \mathbb{R}^{B\times 128}$.

(3) Modeling Dynamic Temporal Dependencies:

The reconstructed sequence X_{seq} is input into a two-layer LSTM branch, with 64 and 128 hidden units in each layer, respectively, to model dependencies between different time units. The hidden state at the last time step of the second LSTM layer is extracted, yielding $F_{LSTM} \in \mathbb{R}^{B\times 128}$.

(4) Contextual Correlation and Local Focus Extraction:

First, a linear embedding layer is used to map each four-dimensional time unit to a 128-dimensional feature space $X_{emb} \in \mathbb{R}^{B\times 256\times 128}$. Subsequently, X_{emb} is fed into a two-layer Transformer encoder comprising RoPE and LWA. RoPE is used to characterize the relative positional relationships between different time units, while LWA computes attention within overlapping local windows. The encoding results are subjected to adaptive average pooling to obtain the contextual correlation features $F_{Trans} \in R^{B\times 128}$.

(5) Feature Layer Fusion and Fault Classification:

The outputs of the three branches are concatenated along the feature dimension to form a composite feature $F_{fusion} = Conca(F_{CNN}, F_{LSTM}, F_{Trans}) \in R^{B\times 384}$. The composite feature vector is then mapped through a fully connected layer and activated by the Softmax function to achieve accurate classification of the 10 typical fault classes of bearings.

4. Experimental Results and Analysis

4.1. Experimental Dataset

This study uses the Case Western Reserve University bearing dataset, which contains vibration measurements from normal bearings and bearings with seeded inner-race, outer-race, and rolling-element faults [16]. The dataset includes vibration signals under normal operating conditions and vibration signals under three different failure conditions; each fault type includes three different failure diameters, resulting in a total of ten distinct categories when combined with the normal state. The composition of the dataset is shown in Table 1:

Table 1. Composition of experimental dataset

Fault Type	Fault Diameter (mm)	Number of Sample	Label
Normal	——	300	1
Inner-Race Fault	0.1778	300	2
	0.3556	300	3
	0.5334	300	4
Outer-Race Fault	0.1778	300	5
	0.3556	300	6
	0.5334	300	7
Rolling-Element Fault	0.1778	300	8
	0.3556	300	9
	0.5334	300	10

A total of 300 samples were constructed for each condition class, resulting in 3,000 samples across the ten classes. To reduce the risk of information leakage caused by overlapping segments, the original continuous vibration recordings were first divided into mutually exclusive training, validation, and test subsets at a ratio of 7:2:1. Sliding-window segmentation was then independently performed within each subset, using a sample length of L=1024 and an overlap ratio of 50% [17]. Consequently, each class contained 210 training samples, 60 validation samples, and 30 test samples. The class distribution was kept balanced across all three subsets, and overlapping segments originating from the same signal interval were prevented from appearing in different subsets.

Since the amplitude of vibration signals generated by the coal cutter's bearings under alternating stress fluctuates dramatically, to eliminate the impact of physical dimension differences on feature extraction in multi-channel parallel networks, this paper employs a min–max (Min-Max) linear normalization algorithm at the model input to scale the original discrete sample points, strictly compressing the dynamic range of the input data to the interval [0, 1]. The calculation formula is as follows:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{12}$$

Where x represents the original actual vibration amplitude of the input sample; x_{max} and x_{min} represent the maximum and minimum values, respectively, in the current analysis sample sequence; x' represents the normalized feature value obtained after dimensionless interval mapping.

4.2. Evaluation Metrics and Hyperparameter Optimization

To objectively evaluate the performance of CLTrans in the ten-class bearing fault diagnosis task, accuracy, macro-averaged precision, macro-averaged recall, and macro-averaged F1-score were selected as the evaluation metrics. Accuracy measures the proportion of correctly classified samples among all test samples. For precision, recall, and F1-score, each fault class was first treated as the positive class in

a one-versus-rest manner, and the class-wise results were then averaged over all classes.

$$\text{Accuracy} = \frac{\sum_{k=1}^{K} c_{kk}}{\sum_{i=1}^{K}\sum_{j=1}^{K} c_{ij}} \tag{13}$$

Where K is the number of fault classes, which is set to 10 in this study, and c_{ij} denotes the number of samples whose true class is i and predicted class is j.

For the kth class, precision, recall, and F1-score are calculated as follows:

$$P_k = \frac{TP_k}{TP_k + FP_k}, \quad k = 1, 2, \ldots, K, \tag{14}$$

$$R_k = \frac{TP_k}{TP_k + FN_k}, \quad k = 1, 2, \ldots, K, \tag{15}$$

$$F_{1,k} = \frac{2P_k R_k}{P_k + R_k}, \quad k = 1, 2, \ldots, K. \tag{16}$$

Here, TP_k denotes the number of samples correctly predicted as class k, FP_k denotes the number of samples from other classes incorrectly predicted as class k, and FN_k =denotes the number of class-k samples incorrectly predicted as other classes.

The key hyperparameters were systematically optimized using grid search. The optimal configuration ultimately determined is shown in Table 2:

Table 2. Key Model Hyperparameters and Settings

Hyperparameters	Optimization Range	Optimal Value
Number of CNN Convolution Blocks	[1, 2, 3]	2
CNN Output Channels	{[32, 64], [64, 128], [128, 256]}	[64, 128]
Number of LSTM Layers	[1, 2, 3]	2
LSTM Hidden Units	{[32, 64], [64, 128], [128, 256]}	[64, 128]
Number of Transformer Encoder Layers	[1, 2, 3]	2
Number of Transformer Attention Heads	[2, 4, 8]	2
LWA Window Size/Step Size	Fixed Configuration	64/32
Learning Rate/Optimizer	[0.01, 0.001, 0.0003, 0.0001]	0.0003/ Adam

The CNN adopts an architecture with two convolutional blocks, with the number of output channels per layer being 64 and 128, respectively. This configuration ensures the ability to distinguish local features while achieving downsampling in the temporal dimension. The LSTM employs a two-layer structure with a hidden unit count of [64, 128]; the improved Transformer is configured with two encoder layers and two attention heads, with an LWA window size and stride of 64 and 32, respectively. The network was optimized using Adam with an initial learning rate of 0.0003 [18], and its weights were initialized using the Xavier initialization scheme [19].

4.3. Diagnostic Process and Analysis of Results

Classification validation was performed using the bearing dataset from Case Western Reserve University (CWRU) in the United States; the model training curve is shown in Figure 4.

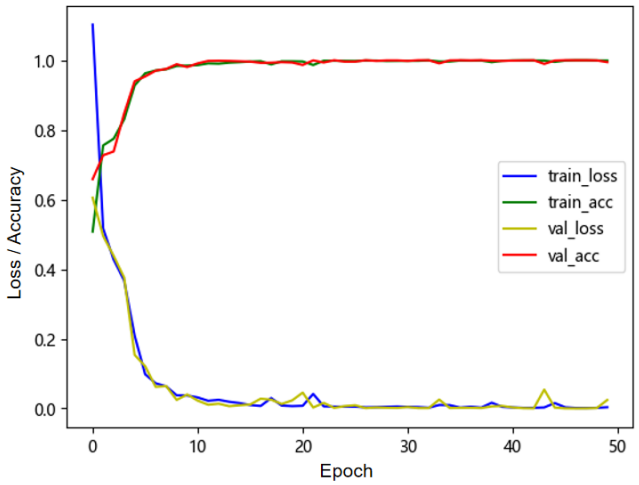


Figure 4. Training and validation curves of the proposed model

As shown in Figure 4, the training loss and validation loss decrease rapidly during the first 10 training iterations and then gradually stabilize; the training accuracy and validation accuracy follow similar trends and do not diverge significantly. The results indicate that CLTrans exhibits relatively stable convergence performance under the current data split and does not show obvious signs of overfitting.

4.4. Comparative Experiments

To evaluate and validate the superiority of the model proposed in this paper, this study selected five models widely used in the field of fault diagnosis for comparative experiments, including a model combining a convolutional neural network with an attention mechanism (CNN-Attention), an empirical mode decomposition-cascaded Transformer model (EMD-Transformer), a model combining Long Short-Term Memory (LSTM) with attention mechanisms (LSTM-Attention), a combined Empirical Mode Decomposition–Convolutional Neural Network–Long Short-Term Memory model (EMD-CNN-LSTM), and a model combining Transformer with bidirectional Long Short-Term Memory (BiLSTM). The comparisons were conducted on the same dataset, and the experimental results are shown in Table 3 and Figure 5:

Table 3. Comparison of fault diagnosis results among different algorithms

Model	Precision	Recall	F1 Score	Accuracy
CNN-Attention	0.8889	0.8879	0.8853	0.8929
EMD-Transformer	0.9415	0.9222	0.9276	0.9286
LSTM-Attention	0.9324	0.9350	0.9333	0.9375
EMD-CNN-LSTM	0.9681	0.9604	0.9634	0.9643
Transformer-BiLSTM	0.9868	0.9714	0.9764	0.9828
CLTrans	0.9955	0.9929	0.9940	0.9957

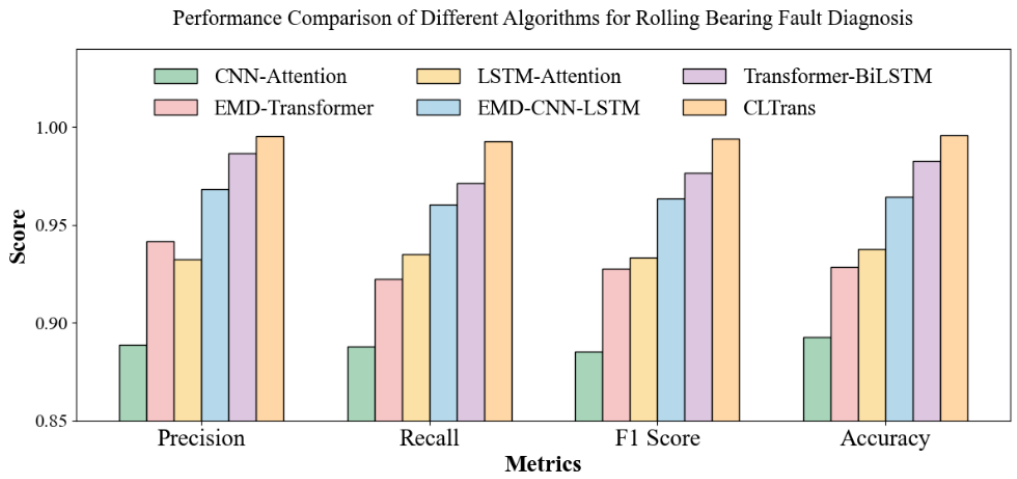
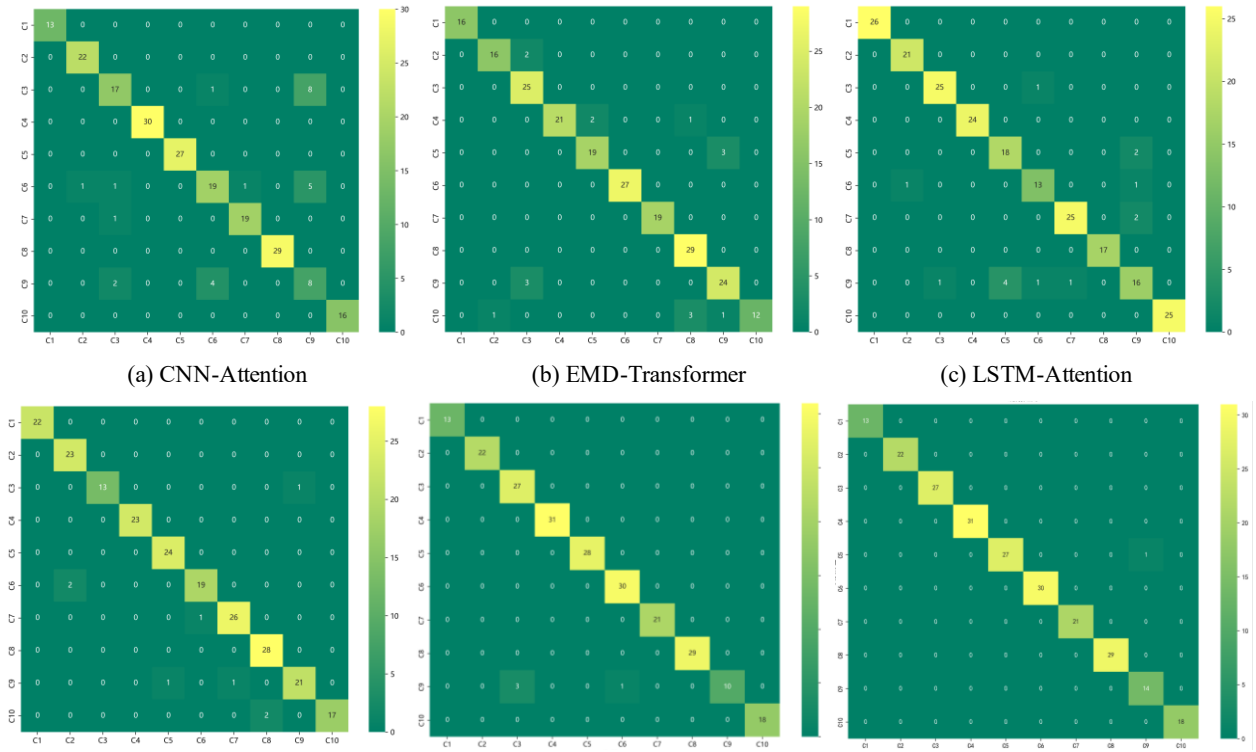


Figure 5. Comparison of bearing fault diagnosis performance of different models

CLTrans achieved the highest results in terms of accuracy, precision, recall, and F1 score. Compared to the best-performing baseline model, Transformer-BiLSTM, CLTrans improved accuracy from 98.28% to 99.57%, precision from 98.68% to 99.55%, and recall from 97.14% to 99.29%, and the F1 score increased from 97.64% to 99.40%, representing improvements of 1.29, 0.87, 2.15, and 1.76 percentage points, respectively.



(a) CNN-Attention (b) EMD-Transformer (c) LSTM-Attention

(d) EMD-CNN-LSTM (e) Transformer-BiLSTM (f) CLTrans

Figure 6. Confusion matrix

To quantitatively evaluate the CLTrans model's performance in identifying different damage categories and degradation levels in rolling bearings at the micro-decision level, this paper plots the 10-class confusion matrices for the mainstream comparison networks and the proposed model; the results are shown in Figure 6. A comparative analysis reveals that traditional CNN-Attention, EMD-Transformer, and LSTM-Attention models exhibit a large number of misclassifications and missed diagnoses on both sides of the diagonal, with discrimination accuracy dropping significantly, particularly at the boundaries between inner-race fault and rolling element damage—areas prone to confusion. In contrast, the prediction results of the CLTrans parallel fusion model proposed in this paper (Figure 6(f)) are primarily concentrated along the main diagonal, with relatively few misclassifications between categories. This indicates that the combined feature representation helps improve the ability to distinguish between different failure locations and damage sizes.

4.5. Ablation Study

To analyze the impact of different feature extraction branches and the fully improved model on diagnostic results, we constructed and compared the CNN-Transformer, LSTM-Transformer, CNN-LSTM, CNN-LSTM-Transformer, and full CLTrans models. The experimental results are shown in Table 4 and Figure 7.

Table 4. Comparison of fault diagnosis results after removing different components

Model	Precision	Recall	F1 Score	Accuracy
CNN-Transformer	0.9180	0.9230	0.8988	0.9062
LSTM-Transformer	0.9587	0.9566	0.9545	0.9598
CNN-LSTM	0.9673	0.9611	0.9628	0.9732
CNN-LSTM-Transformer	0.9887	0.9786	0.9822	0.9871
CLTrans	0.9955	0.9929	0.9940	0.9957

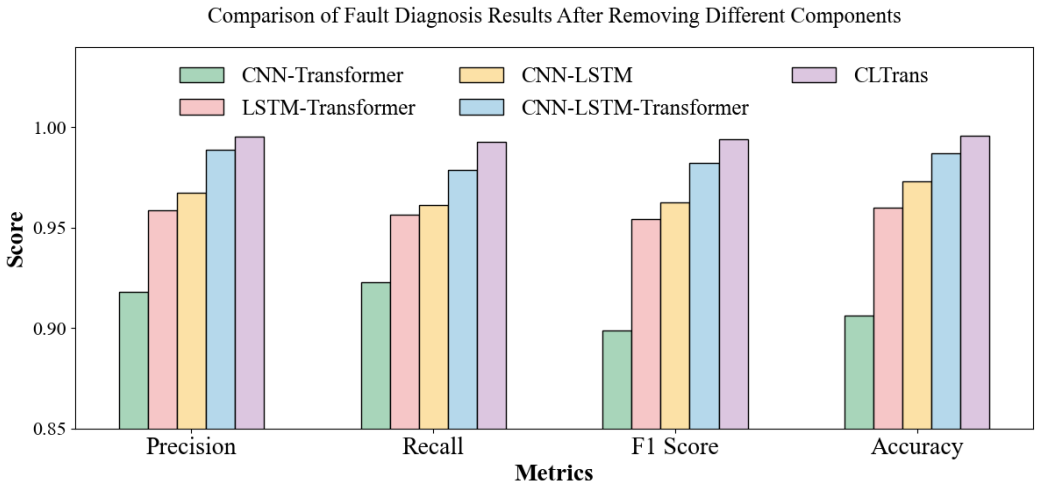


Figure 7. Importance analysis of different components on the model

Overall, the ablation experiments demonstrate that, compared to the CNN-LSTM-Transformer model without RoPE and LWA, the full CLTrans model achieves an accuracy of 99.57% (up from 98.71%) and an F1 score of 99.40% (up from 98.22%), indicating that the improved Transformer module incorporating RoPE and LWA can further enhance the joint feature representation.

5. Conclusion

This paper proposes CLTrans, a rolling bearing fault diagnosis model that integrates CNN, LSTM, and an enhanced Transformer in parallel. The model uses a one-dimensional CNN to extract local transient impact features, employs a two-layer LSTM to model temporal dependencies, and utilizes a Transformer branch that combines RoPE and LWA to extract contextual correlation features. These three feature streams are concatenated and used for fault state classification.

Experimental results on the CWRU dataset show that CLTrans achieves an accuracy of 99.57%, a precision of 99.55%, a recall of 99.29%, and an F1 score of 99.40%. Compared with Transformer-BiLSTM, the accuracy and F1 score are improved by 1.29 and 1.76 percentage points, respectively. Comparison results of different branch combinations indicate that local transient features, temporal dependency features, and contextual correlation features are to some extent complementary.

The current study is primarily validated using publicly available rolling bearing datasets and has not yet fully accounted for random impact noise, load variations, rotational speed fluctuations, and equipment differences in the actual operating environment of coal cutters. Future work will incorporate actual measurement signals from the cutting head of coal cutters to conduct research on cross-operating-condition diagnosis, noise robustness, and lightweight deployment.

References

[1] Randall, R. B., & Antoni, J. (2011). Rolling element bearing diagnostics—A tutorial. Mechanical Systems and Signal Processing, 25(2), 485–520. https://doi.org/10.1016/j.ymssp.2010.07.017

[2] Liu, R., Yang, B., Zio, E., & Chen, X. (2018). Artificial intelligence for fault diagnosis of rotating machinery: A review. Mechanical Systems and Signal Processing, 108, 33–47. https://doi.org/10.1016/j.ymssp.2018.02.016

[3] Jia, F., Lei, Y., Lin, J., Zhou, X., & Lu, N. (2016). Deep neural networks: A promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data. Mechanical Systems and Signal Processing, 72–73, 303–315. https://doi.org/10.1016/j.ymssp.2015.10.025

[4] Wang, H., Liu, Z., Peng, D., & Qin, Y. (2022). Feature-level attention-guided multitask CNN for fault diagnosis and working conditions identification of rolling bearing. IEEE Transactions on Neural Networks and Learning Systems, 33(9), 4757–4769. https://doi.org/10.1109/TNNLS.2021.3060494

[5] Fu, G., Wei, Q., & Yang, Y. (2024). Bearing fault diagnosis with parallel CNN and LSTM. Mathematical Biosciences and Engineering, 21(2), 2385–2406. https://doi.org/10.3934/mbe.2024105

[6] Sun, H., & Zhao, S. (2021). Fault diagnosis for bearing based on 1DCNN and LSTM. Shock and Vibration, 2021, 1221462. https://doi.org/10.1155/2021/1221462

[7] Guo, Y., Mao, J., & Zhao, M. (2023). Rolling bearing fault diagnosis method based on attention CNN and BiLSTM network. Neural Processing Letters, 55(3), 3377–3410. https://doi.org/10.1007/s11063-022-11013-2

[8] Hao, S., Ge, F., Li, Y., & Jiang, J. (2020). Multisensor bearing fault diagnosis based on one-dimensional convolutional long short-term memory networks. Measurement, 159, 107802. https://doi.org/10.1016/j.measurement.2020.107802

[9] Xu, Y., Li, Z., Wang, S., Li, W., Sarkodie-Gyan, T., & Feng, S. (2021). A hybrid deep-learning model for fault diagnosis of

rolling bearings. Measurement, 169, 108502. https://doi.org/10.1016/j.measurement.2020.108502

[10] Ding, Y., Jia, M., Miao, Q., & Cao, Y. (2022). A novel time–frequency Transformer based on self-attention mechanism and its application in fault diagnosis of rolling bearings. Mechanical Systems and Signal Processing, 168, 108616. https://doi.org/10.1016/j.ymssp.2021.108616

[11] Hou, Y., Wang, J., Chen, Z., Ma, J., & Li, T. (2023). Diagnosisformer: An efficient rolling bearing fault diagnosis method based on improved Transformer. Engineering Applications of Artificial Intelligence, 124, 106507. https://doi.org/10.1016/j.engappai.2023.106507

[12] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278–2324. https://doi.org/10.1109/5.726791

[13] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

[14] Su, J., Lu, Y., Pan, S., Murtadha, A., Wen, B., & Liu, Y. (2024). RoFormer: Enhanced Transformer with rotary position embedding. Neurocomputing, 568, 127063. https://doi.org/10.1016/j.neucom.2023.127063

[15] Beltagy, I., Peters, M. E., & Cohan, A. (2020). Longformer: The long-document Transformer. arXiv preprint arXiv:2004.05150.

[16] Smith, W. A., & Randall, R. B. (2015). Rolling element bearing diagnostics using the Case Western Reserve University data: A benchmark study. Mechanical Systems and Signal Processing, 64–65, 100–131. https://doi.org/10.1016/j.ymssp.2015.04.021

[17] Hendriks, J., Dumond, P., & Knox, D. A. (2022). Towards better benchmarking using the CWRU bearing fault dataset. Mechanical Systems and Signal Processing, 169, 108732. https://doi.org/10.1016/j.ymssp.2021.108732

[18] Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In Proceedings of the International Conference on Learning Representations.

[19] Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (Vol. 9, pp. 249–256).