International Journal for Housing Science and Its Applications
Publish August 4, 2025. Volume 46, Issue 3 Pages 1684-1698

https://doi.org/10.70517/ijhsa463131

Research on System Fault Detection and Predictive Maintenance Methods Based on Deep Convolutional Neural Networks and Time Series Analysis

Nan Luo[1,*]
[1] Suzhou Vocational Institute of Industrial Technology, Suzhou, Jiangsu, 215104, China
Corresponding authors: (e-mail: jdluonan@163.com).

Abstract In recent years, China has made great efforts to develop fault diagnostic systems, and predictive maintenance in advance based on operational data can reduce downtime, operational costs and personnel safety in construction projects. In this paper, the time-frequency analysis method of instantaneous Fourier transform is used to study the time-frequency variation of non-smooth signals, and the appropriate window length is selected according to the vibration signals in order to provide better signal preprocessing results. Combine LSTM and CNN to deal with the system fault classification task, and introduce CBAM attention mechanism on this basis to improve the ability of recognizing and classifying fault signals in vibration data. Construct a predictive maintenance model to perform predictive maintenance on the system. The time-domain signals of severe faults are different from the other two states, and the amplitude is close to 10 $m \cdot s^{-2}$, which is easy to judge the system state after deep convolutional neural network calculation. Meanwhile, the predictive maintenance model designed in this paper incorporates the temporal convolutional network model with attention mechanism, which improves 7.042% and 45.223% in scores compared with the long memory network and bidirectional LSTM, which is of great value for practical system lifetime prediction.

Index Terms instantaneous Fourier transform, CNN, CBAM, predictive maintenance

I. Introduction

Maintenance of equipment is an important process in the production chain, equipment maintenance methods in accordance with the implementation of different equipment maintenance measures, the enterprise will produce different economic benefits [1], [2]. In particular, predictive maintenance is a maintenance approach that develops a maintenance plan based on the current state of operation of the equipment and future degradation trends, which provides a maintenance strategy for the equipment in advance and supports the full lifecycle management of the equipment [3]-[5]. In addition, it allows the application of predictive tools on a wider range of equipment through the use of predictive tools such as artificial intelligence methods, integrity factors, statistical inference methods, and engineering techniques to improve fault detection and maintenance efficiency [6]-[8]. Predictive maintenance effectively avoids economic losses due to large-scale failures and has many advantages such as improving equipment reliability, reducing maintenance costs, and prolonging equipment service life, which are widely used by enterprises [9], [10]. With the development of artificial intelligence technology, especially the progress of machine learning and deep learning, fault detection has ushered in a new direction of improvement [11], [12]. Intelligent fault detection methods can not only improve the speed and accuracy of detection, but also predict possible future faults by learning historical data, which is important for maintaining the stable operation of the system and preventing large-scale failures [13]-[16].

In the field of fault detection, Support Vector Machines (SVMs) are capable of constructing classification models by training historical data to make binary or multi-classification judgments of system operation states.Li, J. et al. proposed an SVM-based HVAC fault detection and diagnosis model, which learns the consistent nature of equipment operation faults and then realizes the automated detection and diagnosis of the faults [17]. And reinforcement learning realizes the real-time monitoring and optimization of the system operation state by constructing the interaction relationship between the intelligent body and the environment.Mahmood, T. et al. applied reinforcement learning technology to the intelligent fault detection system for wireless sensor networks, which significantly improves the robustness of the network through the real-time adaptation to the network environment, which is conducive to the reduction of the network failure risk [18]. In addition, a decision tree is a model that represents decision rules in a tree structure, by dividing the feature space layer by layer and finally realizing the classification task.Benkercha, R. and Moulahoum, S. constructed a fault detection and diagnostic model based on the decision tree algorithm for grid-connected photovoltaic systems, which adopts different numerical attributes and objectives as the basis of classification to achieve a high detection and diagnostic accuracy [19]. The application of the above machine learning-based fault detection methods in the field of fault detection accurately identifies and predicts potential faults in equipment and systems with good adaptability and practicality.

Big data platforms also provide efficient data storage and computational capabilities for fault detection, especially in the face of massive log and sensor data. Among them, real-time data processing is a key link in fault detection.Calabrese, F. et al. utilized stream processing and incremental clustering-based methods to label massive data in information systems, thus providing timely feedback on the health of the system based on the contextual information, and providing a strong guarantee for efficient fault detection [20]. Data mining techniques, on the other hand, discover potential fault occurrence patterns through pattern mining and correlation analysis of historical data. For example, Xue, P. et al. introduced the potential application of data mining technique in centralized heating system, which can extract valuable knowledge from a large amount of substation operation data and provide data support for system fault detection and operation optimization [21]. The above failure prediction methods based on big data analysis play an important role in real-time data analysis for system failure detection by virtue of their excellent memory computing and batch processing capabilities.

In recent years, some scholars have also applied deep learning in the field of predictive maintenance to establish deep learning models suitable for predictive maintenance. Among them, convolutional neural network (CNN) has significant advantages in image and high-dimensional data processing. Wen, L. et al. showed that traditional data-driven fault detection and diagnosis methods are highly dependent on inefficient expert extraction of features, for this purpose a novel CNN network based on LeNet-5 is established for fault diagnosis, which exhibits good adaptivity and prediction accuracy [22]. Kiranyaz, S. et al. designed a one-dimensional convolutional neural network without the need for a feature extraction process and applied it to a modular multilevel converter circuit detection system, which not only achieves efficiency in the detection of abnormal voltage and current data, but also improves the reliability of fault identification [23]. And Recurrent Neural Networks (RNN) are widely used in the analysis of time series data due to their ability to memorize historical information.Shahnazari, H. used RNN networks for modeling and inversion of nonlinear systems, which do not require inputs of historical fault information and are capable of isolating highly interconnected actuators and sensors in the system to improve the efficiency and accuracy of fault detection [24]. Zhang, Y. et al. combined an RNN network with a gated loop unit to convert one-dimensional time-series vibration signals into two-dimensional images and extract fault features from them, thus realizing robust detection and identification of mechanical fault types [25]. Based on this, the combination of deep convolutional neural network and time series analysis can better capture the uncertain data in the system, and has more prominent advantages than machine learning techniques and big data techniques in dealing with complex failure modes, which is worthy of further research.

In this paper, the time-frequency variation of fault signals is obtained in a short period of time by sliding a fixed-size window and applying a Fourier transform to its position. A combination of LSTM and CNN models is used to enhance the performance of the model in system fault classification tasks. The introduction of the CBAM attention mechanism ensures that the model effectively captures the dynamic characteristics of the time-frequency features, and improves the ability to recognize and classify fault signals in vibration data. The model is experimentally verified to capture the time domain and frequency domain signals of faults within the system. On the basis of the LSTM model, a recurrent neural network is added to specifically process the sequence data, and the predictive maintenance model is designed using the XGBoost objective function. Predictive performance is analyzed by controlled experiments on production system fault state prediction. The model is applied to actual engineering analysis to evaluate the specific use effect of its fault detection and predictive maintenance.

II. System fault detection model based on deep convolutional neural network

II. A.Deep Learning Fundamentals

II. A. 1) Time-frequency conversion

The transient Fourier transform (STFT) is a time-frequency analysis method that is commonly used to analyze and study the time-frequency variations of non-stationary signals [26]. By sliding a fixed-size window and applying the Fourier transform to its position, the STFT can obtain a localized spectrum in a short period of time, overcoming the limitation that the conventional Fourier transform is unable to provide time-localized information.The STFT is able to decompose the signal into different time points and frequency components, generating a two-dimensional time-frequency diagram. The amplitude of each point in the time-frequency diagram reflects the signal strength at a specific time and frequency, and its mathematical expression is:

$$STFT\{x(t)\}(\tau,\omega)=\int_{-\infty}^{+\infty}x(t)\omega(t-\tau)e^{-j2\pi\omega t}dt \tag{1}$$

where: $x(t)$ - time domain signal, τ - time position parameter reflecting the frequency characteristics of the signal at different time periods, ω - the frequency variable, indicating the analysis of the characteristic frequency, $\omega(t-\tau)$ - a window function around the time τ, $e^{-j2\pi\omega t}$ - is the kernel of the Fourier Transform function.

Commonly used window functions include Hamming windows, rectangular windows, and triangular windows. Compared with other window functions, the Hamming window has stronger suppression of the secondary flap and moderate width of the primary flap, which can balance the time and frequency resolution in spectrum analysis. The smoother shape of the Hamming window helps to minimize the time-domain leakage effect and preserve the signal detail information. The Hamming window function is:

$$\omega(n)=0.54-0.46\cos\left(\frac{2\pi n}{N-1}\right),0<n<N-1 \tag{2}$$

where: n - the ordinal number of the window function, N - the length of the window.

However, the resolution of the STFT depends on setting the window size. The window length in turn determines the spectral time and frequency resolution, and the time resolution is:

$$T=[(N_x-N_o)/(N_s-N_o)] \tag{3}$$

where: [] - the operation of rounding down, N_x - the length of the signal ready to be processed, N_o - the window function transformation overlap length during the process, N_w - length of the window function. The frequency resolution is defined as:

$$F=\begin{cases}N_x/(2+1)\\(N_x+1)/2\end{cases} \tag{4}$$

Longer windows provide higher frequency resolution after Fourier transform, so the appropriate window length should be selected according to the vibration signal to provide better signal preprocessing results.

II. A. 2) Long and short-term memory networks

In system fault classification tasks, LSTM can be used in combination with CNN to improve model performance [27], [28]. In this combination, CNN is used to extract features from vibration signals and LSTM is used to process sequence data and capture its time dependence.CNN is responsible for extracting local features and LSTM is responsible for overall sequence modeling, and the combination of the two can handle the system fault classification task more efficiently.The specific process of extracting features in the LSTM layer is shown in Eqs. (5) to (9):

$$i_t=\delta(W_i\times[h_{i-1},x_i^{'}]+b_i) \tag{5}$$

$$f_t=\delta(W_t\times[h_{t-1},x_t^{'}]+b_t) \tag{6}$$

$$o_t=\delta(W_o\times[h_{t-1},x_t^{'}]+b_o) \tag{7}$$

$$C_t=f_t\times C_{t-1}+i_t\times\tanh(W_c\times[h_{t-1},x_t^{'}]+b_c) \tag{8}$$

$$h_t=o_t\times\tanh(C_t) \tag{9}$$

where: i_t - input gate at time t, f_t - forgetting gate at time t, o_t - - output gate at time t, C_t - memory cell output at time t, $x_t^{'}$ - input vector, h_{t-1} - hidden layer vectors, δ - Sigmoid activation functions, W_i,W_f,W_o,W_c -Weighting coefficients of h_{t-1} for forgetting gate, input gate, output gate and feature extraction process, b_i,b_f,b_o,b_c - -Oblivion gates, input gates, output gates and weights in the feature extraction process.

II. A. 3) Convolutional neural network structure

CNN is the core architecture in deep learning, especially effective in processing 2D images.The convolutional layer of CNN extracts the features of the input data through convolutional operations, and enhances the nonlinear representation of the network through activation functions (e.g., ReLU). The pooling layer retains the key features by reducing the spatial dimensions of the feature map, and the fully connected layer maps the output of the pooling layer to the classification result and achieves multi-category classification through the Softmax function. By extracting the automated features of the CNN, it reduces the need for manually designing features in traditional image processing techniques and is robust to the location of the input image.

When the input to the CNN is a vibration signal time-frequency map, the feature representation is achieved after convolutional, pooling and fully connected layers. For a 2D input X and a 2D convolution kernel K, the formula for the convolution operation (without considering boundary effects and step size) is as follows:

$$(X^{*}K)(i,j)=\sum_{m}\sum_{n}X(i+m,j+n)K(m,n) \tag{10}$$

where: $(X*K)(i,j)$ - convolution response at position (i,j), * - convolution, (i,j) - coordinate positions on the output feature map, the m and n - coordinate positions within the convolution kernel.

After the convolution operation, the pooling layer is used to reduce the spatial dimensionality of the output of the convolution layer while retaining the key information. Pooling is divided into maximum value pooling and average value pooling. Maximum pooling is usually used, and its formula is as follows:

$$p_j=\max\nolimits_{(m,n)\in window} x_{i,j}^{l}(m,n) \tag{11}$$

where: p_j - output after pooling, max - select the maximum value, $x_{i,j}^{l}(m,n)$ - the l th level of the value of the i th feature map at position (m,n), window - perform pooling operation neighborhood region.

At the back of the network, the features learned in the previous layers are pooled and classified through the fully connected layer y_j. The fully connected layer operates with the following formula:

$$y_j=f\left(\sum_{i}w_{i,j}x_i+b_j\right) \tag{12}$$

where: $f()$ - activation function such as sigmoid, tanh or ReLU, $w_{i,j}$ - weight between the i th input and the j th output in the fully connected layer, x_i - the i th input of the previous layer, b_j - the bias term of the j th output.

II. A. 4) Attention mechanisms

CBAM is a widely used attention mechanism in deep learning, which works by weighting the channel and spatial attention of the feature map at the convolutional block level to enhance the model's perception of key features [29]. Compared to the traditional attention mechanism, CBAM is able to focus on both the channel and spatial dimensions of the feature map, capturing the correlation and importance between features more comprehensively. By introducing the CBAM attention mechanism, the model is able to capture the dynamic characteristics of the time-frequency features more effectively, and improve the ability to recognize and classify fault signals in vibration data. Figure 1 shows the CBAM network structure.

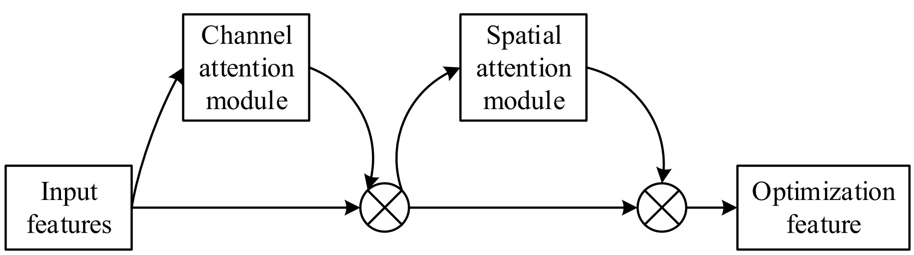


Figure 1: CBAM network structure diagram

One-dimensional channel attention focuses on the importance of each channel in the input feature map. The weights of each channel are computed by performing a global pooling operation (e.g., global average pooling or global maximum pooling) on each channel with a series of fully connected layers and activation functions. These weights are used to adjust the feature response of each channel to enhance the feature representation of the important channels and weaken the feature representation of the unimportant channels. The formula is as follows:

$$F^{'}=A_c(F)\otimes F \tag{13}$$

$$A_c(F)=\delta(F_c(AvgPool(F))+F_c(MaxPool(F))) \tag{14}$$

where: F - input features, F' - output after channel attention module, $A_c(F)$ - channel attention weights, $\otimes$ - element multiplication, δ - Sigmoid activation function, AvgPool - mean pooling, MaxPool - maximum pooling, F_c - shared network of convolution-motivation-convolution structures in the channel attention module.

II. B.Experimental validation

II. B. 1) Fault internal time and frequency domain analysis

Case Western Reserve University system experimental data were used. The drive end bearing model is SKF6205, and the deep convolutional neural network is used to determine the system fault, the depth of the fault is 0.2746 mm, the diameter of the fault is 0.1758 mm, and the sampling frequency is 12,000 Hz. The time domain graph of the system fault signal is shown in Fig. 2, and the amplitude of the time domain of the fault is between -4 and 4 in the experimental time of 0.1 s. The time domain amplitude of the fault is between -4 and 4 in the experimental time.

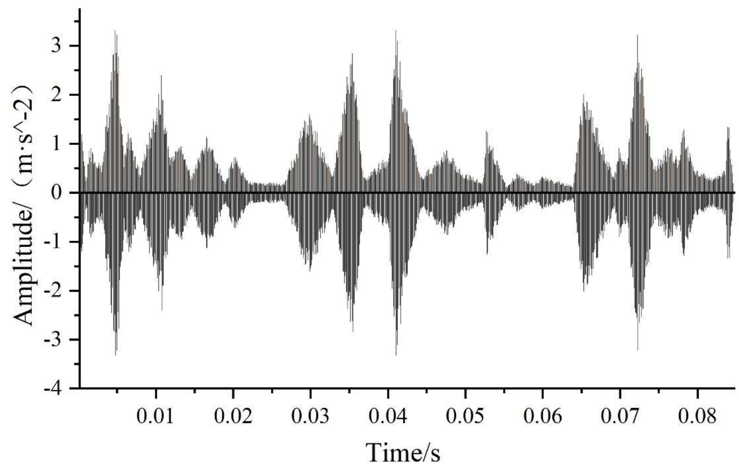


Figure 2: Time domain of system failure signal

Fig. 3 shows the internal fault frequency domain of the system, the instantaneous Fourier transform exists with high temporal resolution, and four more prominent peaks appear in the frequency domain, the first peak appears around 500 Hz, with an amplitude of 0.125m·s^{-2}, the second peak is at 2815 Hz, with an amplitude of 0.325 m·s^{-2}, and the third and fourth at 3000 Hz and 3250 Hz with amplitudes around 0.25 m·s^{-2}. The instantaneous Fourier transform has a high temporal resolution.

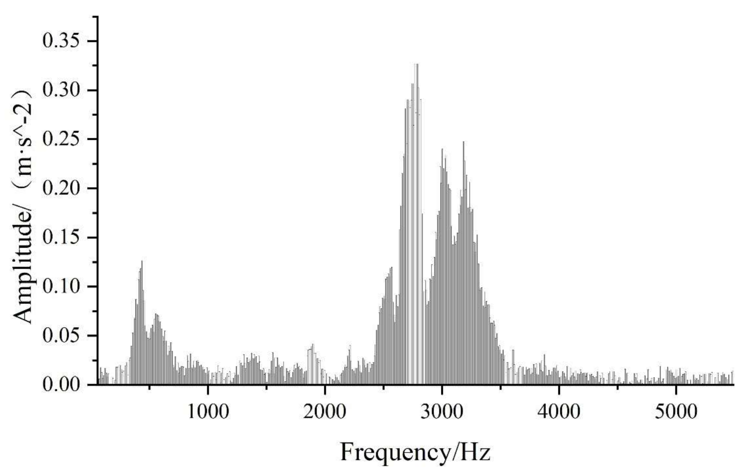
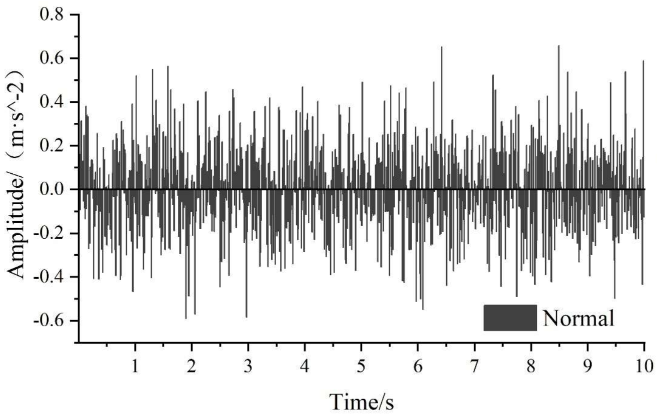


Figure 3: System internal fault frequency domain

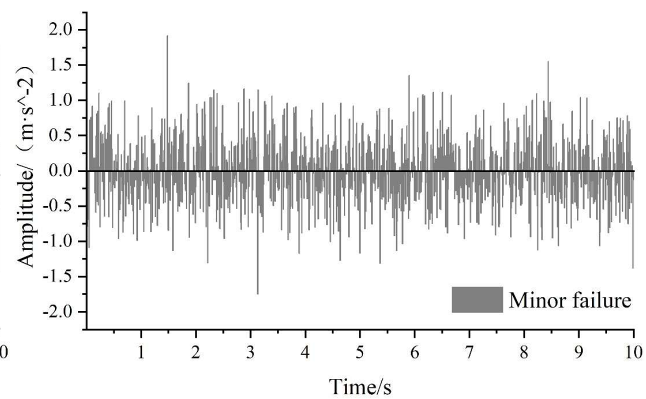
II. B. 2) Time-domain signals in different states

Different errors are repeatedly applied to the system to simulate the fault situation.The sampling frequency in the experiment is 50 kHz and 200 data samples are collected for each state.80% of the data samples are randomly selected as training samples and the remaining as test samples.By constructing the sample set, there are 600 samples for different states, of which 480 are used to form the training samples and 120 are used as test samples.

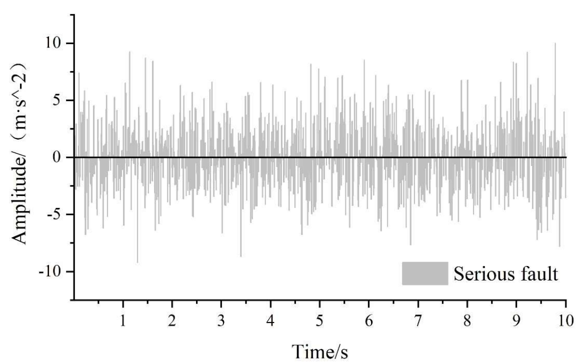
According to the degree of wear and tear is defined as different labeled values: Normal is 1, minor failure is 2, and severe failure is 3. The time domain signals of the different states of the system are shown in Fig. 4, Fig. (a) is the normal state, Fig. (b) is the minor failure, and Fig. (c) is the severe failure. The amplitude of the time-domain signals of the normal state ranges from -0.65m·s^{-2}~0.65m·s^{-2}, the amplitude of the time-domain signals of the minor fault ranges from [-2,2] m·s^{-2}, and the time-domain signals of the severe fault are different from the other two states, with an amplitude close to 10 m·s^{-2} it is easy to judge the system state by the model designed in this paper.



(a)Normal



(b)Minor failure



(c)Serious fault

Figure 4: Time domain signals in different states of the system

III. Predictive maintenance of system failures based on deep time-domain analysis

III. A. Timing prediction model

III. A. 1) Deep learning time series models

(1) Recurrent Neural Network

Based on the LSTM model, the recurrent neural network model is introduced, which is specialized for processing sequence data, and is designed to capture the temporal dependencies in the data.The basic principle of RNN lies in the introduction of recurrent connections, which enables the network to retain the information of the previous moments when processing inputs at each moment.The basic structure of RNN consists of an input layer, a hidden layer, and an output layer. Different from traditional neural networks, RNN introduces the concept of time step between hidden layers, and transfers the hidden state of the previous moment to the current moment through cyclic connections to realize the modeling of sequence information. As shown in Fig. 5, at each time step, the RNN receives the input of the current moment and the hidden state of the previous moment, and then calculates the hidden state of the current moment. The hidden state contains the input information of the current moment and the information of the previous moment.

The mathematical expression of RNN is as follows:

$$h_t = \tanh(W_{ih}x_t + b_{ih} + W_{hh}h_{t-1} + b_{hh}) \tag{15}$$

where W_{ih} and b_{ih} are weight matrices, b_{ih} and b_{hh} are bias terms, and tanh is the hyperbolic tangent activation function.

The hidden state continues to be passed to the next time step or as an input to the output layer. The whole process can be performed iteratively until all time steps have been processed.

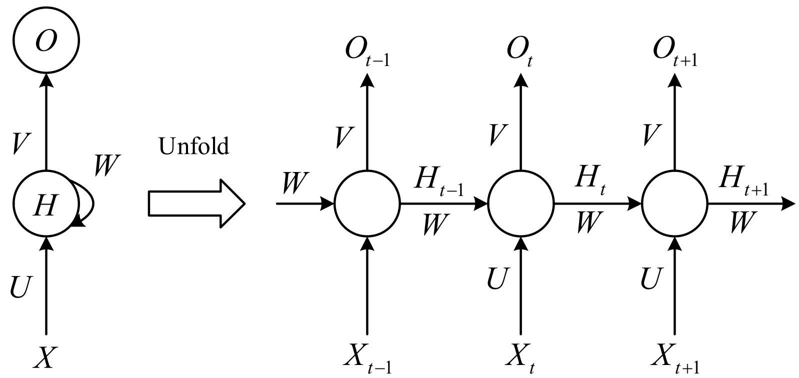


Figure 5: RNN structure diagram

(2) Gated Recurrent Unit

Gated Recurrent Unit (GRU) improves the recurrent neural network and ameliorates the problem of gradient convergence to zero caused by successive products of RNN. Compared to LSTM, GRU improves the computational efficiency of the model by reducing the number of gating units, while maintaining a better long-term dependency modeling capability.

GRU consists of two key gating mechanisms: update gate and reset gate. Its basic structure is as follows:

1) The update gate is computed with the following equation:

$$z_t = \sigma(W_{iz}x_t + W_{hz}h_{t-1} + b_z) \tag{16}$$

The update gate determines how much information needs to be updated for the current time step.

2) The reset gate is calculated with the following formula:

$$r_t = \sigma(W_{ir}x_t + W_{hr}h_{t-1} + b_r) \tag{17}$$

The reset gate determines the effect of the hidden state of the previous time step at the current time step.

3) The new state is computed with the following equation:

$$\tilde{h}_t = \tanh(W_{ih}x_t + r_t \odot (W_{hh}h_{t-1}) + b_h) \tag{18}$$

The new state is calculated by considering the reset gate.

4) The hidden state is updated with the following formula:

$$h_t = (1 - z_t) \odot \tilde{h}_t + z_t \odot h_{t-1} \tag{19}$$

The hidden state smoothly blends the new state with the old state by updating the gate.

III. A. 2) General structure of the forecasting model

(1) Input Layer: The input layer of the model receives key characteristics of each stage of the failure process of the production machine system, which include but are not limited to.

1) Material selection and quality.

2) Process parameters, such as gelatin to water mixing ratio, closure method, color type, plasticizer type, molding temperature and drying humidity.

3) Pass rate of historical production batches.

(2) Convolutional layer: The data were processed by applying a one-dimensional convolutional layer containing 32 convolutional kernels, each of size 3, which were nonlinearly transformed using the ReLU activation function. This step is used to extract local features from the original sequence.

(3) Activation function: the output of the convolutional layer is nonlinearly transformed using the ReLU activation function, which introduces a nonlinear factor that helps the model to learn more complex patterns.

(4) Maximum Pooling Layer: A one-dimensional maximum pooling operation is performed on the output of the convolutional layer for data reduction and feature extraction. This helps to reduce the number of parameters in the model and facilitates the extraction of the most significant features within the time window.

(5) GRU layer:

The input passes through a GRU layer with hidden_size implicit units. This layer has the ability to memorize and learn long-term dependencies, where hidden_size indicates the number of GRU units.

(6) Self-attention mechanism:

With the self-attention mechanism, the model can focus on important information at different locations in the sequence. The self-attention layer calculates the weight of each time step to better capture the key information in the sequence.

(7) Fully connected layer:

Finally, the output of the GRU layer is mapped to the final output of the model through the fully connected layer. This layer maps the features of the hidden_size dimension to the output size, using the Sigmoid activation function to produce the final prediction.

III. B. Predictive maintenance model design

III. B. 1) Self-encoder

An auto-encoder is an artificial neural network for data encoding for unsupervised learning. An auto-encoder consists of two parts: an encoder function and a decoder function, which compresses the input data into a low-dimensional representation through an encoder and then attempts to reconstruct the original data from this low-dimensional representation through a decoder. The autoencoder learns through efficient training the feasible compression of the data, i.e., the optimal representation of the information passed between the encoder and the decoder, in order to minimize the reconstruction error. This mechanism enables the autoencoder to capture the most important features of the data. Figure 6 shows the entire network structure.

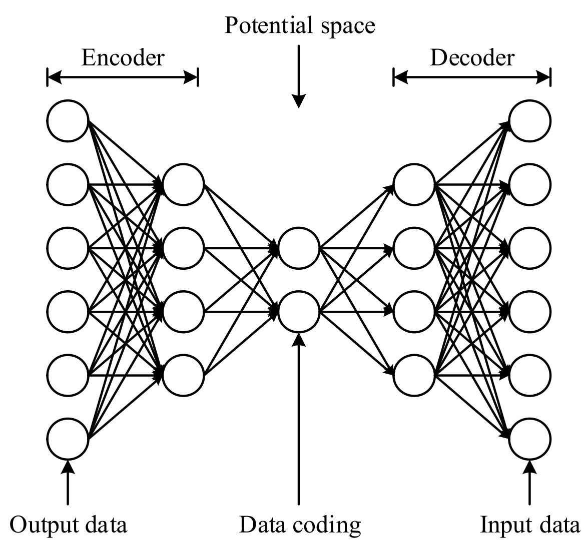


Figure 6: Autoencoder network structure diagram

(1) Encoder

The encoder is responsible for capturing key features in the data and compressing the input data into a low dimensional representation that is used to find the underlying layers of the input:

$$h = f(x) = s(W_x + b) \tag{20}$$

where s is a nonlinear activation function.

(2) Decoder.

The decoder, on the other hand, is responsible for reconstructing the original data from the previous decoding and outputting the output vector of the original reconstruction. The transformation g maps the resulting hidden representation to the d-dimensional reconstruction r of x:

$$r = g(h) = s(W_h^{'} + b^{'}) \tag{21}$$

where s is a nonlinear function, $W^{'}$ is the weight matrix, and $b^{'}$ is the deviation vector.

In the above autoencoder, the coding and structural parameters are $\theta = \{W, b\}$ and $\phi = \{W^{'}, b^{'}\}$, respectively. The parameters of the encoder and decoder can be trained together to minimize the objective function:

$$\begin{aligned} \theta^*, \phi^* &= \arg\min \frac{1}{n}\sum_{i=1}^{n} L(x^i, r^i) \\ &= \arg\min \frac{1}{n}\sum_{i=1}^{n} L(x^i, g(f(x^i))) \end{aligned} \tag{22}$$

where n is the number of training samples and L is the loss function, which can take a specific form depending on the distribution of the input.

The advantages of self-encoders, as analyzed, include the ability to learn a compressed representation of the data, denoising capabilities, and feature learning. Disadvantages are that they may be difficult to train (especially deep networks), overfitting, and limited representation of complex data structures.

III. B. 2) XGBoost Algorithm

(1) XGBoost objective function

The objective function of XGBoost algorithm is as follows:

$$Obj = \sum_{i=1}^{n} l(\hat{y}_i, y_i) + \sum_{k=1}^{k} \Omega(f_k) \tag{23}$$

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \| w \|^2 \tag{24}$$

The objective function of the XGBoost model consists of two parts, the empirical loss and the structural loss, where l is the tree model empirical loss function, and there are two main types of squared loss function and regression loss function. Ω is the regression tree regularization term.

(2) XGBoost tree model tth assuming that the tth tree is a binary tree, there are:

$$\hat{y}_i^{(t)} = \sum_{k=1}^{t} f_k(X_i) = \hat{y}_i^{(t-1)} + f_t(X_i) \tag{25}$$

where $\hat{y}_i^{(t)}$ is the value of the function after the t th iteration, $\hat{y}_i^{(t-1)}$ is the value of the function after the $t-1$ th iteration, and $f_t(X_i)$ is the functional form of the t th tree.

(3) Taylor's second order expansion

Substituting Eq. (25) into Eq. (24), its second order Taylor expansion is obtained:

$$Obi^{(t)} \approx \sum_{i=1}^{n} \left[l(y_i, \hat{y}_i^{(t-1)}) + g_i f_t(X_i) + \frac{1}{2} h_i f_t^2(X_i) \right] + \Omega(f_t) + cons \tag{26}$$

That is, the optimized function is obtained:

$$Obi^{(t)} \approx \sum_{i=1}^{n} \left[l(y_i, \hat{y}_i^{(t-1)}) + g_i f_i(X_i) + \frac{1}{2} h_i f_t^2(X_i) \right] + \Omega(f_t) \tag{27}$$

where g_i and h_i are the first-order and second-order partial derivatives of the loss function with respect to $\hat{y}^{(t-1)}$, respectively.

(4) Definition of decision tree and its complexity

The XGBoost tree model consists of weight vectors of leaf nodes and mapping relationships of instance leaf nodes, both of which form the branching structure of the tree. The complexity of the tree is represented as follows:

$$\Omega(f_t) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{r} W_j^2 \tag{28}$$

(5) Leaf node grouping

Denote all the samples of the j th node by I_j, and substitute Eq. (27) and Eq. (26) into Eq. (25) to obtain:

$$Obj^{(t)} \approx \sum_{j=1}^{T}\left[\left(\sum_{i\in I_j} g_i\right)W_j + \frac{1}{2}\left(\sum_{i\in I_j} h_i + \lambda\right)W_j^2\right] + \gamma T \tag{29}$$

Define G_j and H_j as the sum of the first-order partial derivatives and second-order partial derivatives of all the samples of this leaf node, respectively, and optimize the node weights to obtain the final optimal objective value after substituting into Eq. (29):

$$Obj^{(t)} = -\frac{1}{2}\sum_{j=1}^{T}\frac{G_j^2}{H_j + \lambda} + \gamma T \tag{30}$$

III. C. Forecast results and analysis

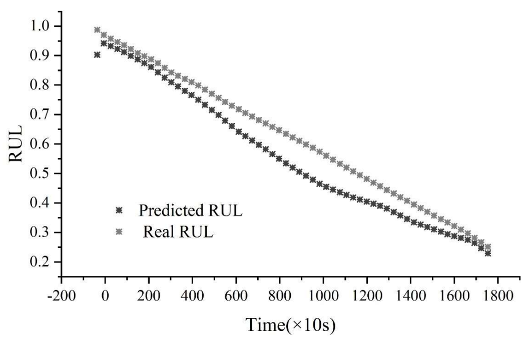
III. C. 1) Comparison of Lifetime Prediction Results

In order to verify the effectiveness of the deep learning time series model to predict the system lifetime, in this paper, 12 cumulative features are input into the XGBoost model, and the system lifetime ratio is used as the label of the data. The optimization algorithm used in this chapter is RNN, and its learning rate is set to 0.0001. The experiments are carried out under the environment of AMDR7-4800Hcpu with RTX2060 graphics card, and the deep learning framework used is Tensorflow2.0.

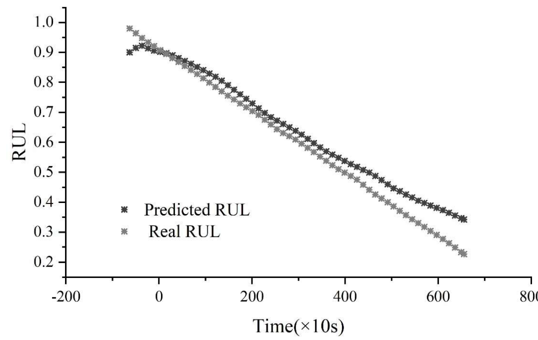
Table 1: The results of system life prediction are compared

System	Check time	Actual	Forecast	Er%				
				This article	Long length memory network	General model	Feature extraction	Two-way LSTM
1	18460	5742	7426.972	6.485	0.348	54.763	-1.048	-5.248
2	11340	2913	4562.398	8.516	5.626	38.632	85.866	-8.765
3	23010	1626	9257.564	23.648	100	-99.455	-278.498	22.364
4	23010	1468	9257.564	-29.948	28.069	-120.148	19.148	-17.823
5	15020	7515	6042.965	9.565	-19.535	70.648	-7.165	43.218
6	12020	7500	4835.981	10.148	-20.178	75.523	10.452	44.835
7	6150	1348	2474.316	-72.389	8.631	19.817	51.826	32.849
8	20030	3048	8058.627	24.864	23.385	8.245	28.848	49.786
9	5790	1196	2329.478	49.846	58.932	17.862	-20.948	25.745
10	1750	516	704.0738	-105	5.168	1.648	44.836	-41.094
11	3540	866	1424.241	6.348	40.248	2.965	-3.644	19.753
|Er|				31.248	28.189	46.384	50.154	28.348
Score				0.456	0.426	0.378	0.359	0.314

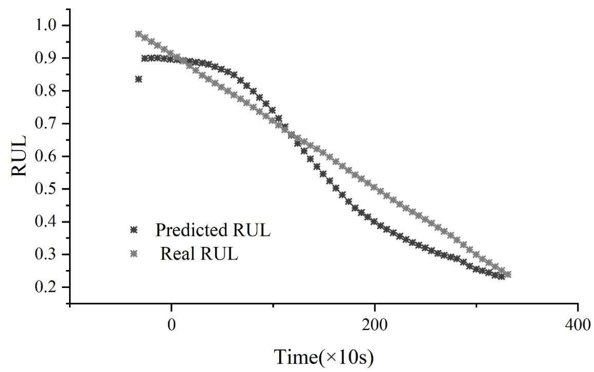
In order to verify the advantages of the improved temporal convolutional network model incorporating the attention mechanism in this paper, the paper is compared with the traditional time-frequency domain features for predicting the system lifetime as well as the traditional temporal neural network for predicting the lifetime. A comparison of the results of system life prediction is shown in Table 1. One system is selected for each working condition to visualize the prediction results, and the prediction results are shown in Fig. 7, and Figs. (a)-(c) are for working conditions 1-3. Taking system 7 as an example, the non-full-life cycle of system 7 is 615 groups, and the time of each group is 10 s. This paper predicts that the ratio of the remaining life is 0.2869, and predicts that the remaining life of the system is 6150/(1-0.2869)-6150= 2474.316s, then its error is (1348-2474.316)/1348=-83.555%. The method in this paper achieved a high score (0.456), and the temporal convolutional network model incorporating the attention mechanism in this chapter improved in score by 7.042% compared to the long memory network and bidirectional LSTM by 45.223%. The method designed in this paper has relatively low percentage error and highest score compared to other methods. The method in this paper provides relatively accurate RUL prediction, which is valuable for practical system lifetime prediction.



(a)Operating condition 1



(b)Operating condition 2



(c)Operating condition 3

Figure 7: The system prediction results under different operating conditions

III. C. 2) Control experiments

In this paper, LSTM+RNN model is used to predict the fault state prediction of production system, in which LSTM network model is used for the prediction of temporal production line state feature vector data, and MLP is used for the classification of the fault state, i.e., the output of the last moment of the LSTM layer is put into the fully-connected layer as an input, and the probability of different categories is obtained by the prediction of the fully-connected layer and the softmax layer. In order to further verify the effectiveness of the proposed method, the LSTM in the prediction model is replaced with RNN structure, and the classification algorithm is replaced with SVM, and the same parameters are used for the experiments, and the MLP and SVM are supervised learning, and their parameters are obtained through the same state-labeled dataset training. Figure 8 shows the prediction results of the four models, LSTM+RNN, LSTM+SVM, RNN+MLP, RNN+SVM. After replacing the hidden layer in the fault prediction model of this paper with RNN, the prediction recall, precision, correctness and the processing time of the prediction model are reduced. Overall, the method proposed in this paper effectively utilizes the feature extraction capability of LSTM for time series data, and in combination with the use of RNN, it effectively achieves the prediction of the fault state of the production line system, and the combined prediction recall, precision, and accuracy of LSTM+RNN are 0.964, 0.945, and 0.964, respectively.

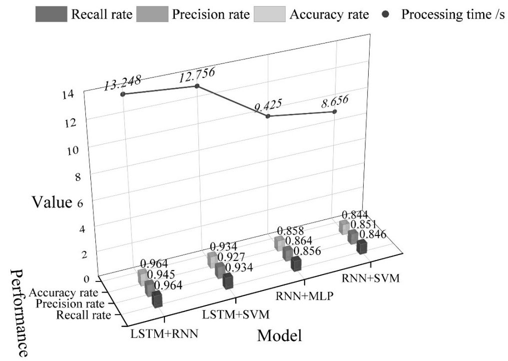


Figure 8: The prediction of the model

IV. Analysis of engineering applications

IV. A. Fault Detection

The acquired raw operational data of the system are time-series transformed, and the evidence before and after the trnsformation is assigned a confidence function using data fusion method respectively. The support of each piece of evidence for the failure modes before the time series transformation is shown in Fig. 9, and the support of each piece of evidence for the failure modes after the time series transformation is shown in Fig. 10, and the failure modes 1-8 are shown in Fig. 9 and Fig. 10 in (a)-(h), respectively.

Combined with the confidence function assignment in Fig. 10 and Fig. 11, respectively, the fusion is performed, and Fig. 11 shows the comparison of the fault confidence values of the fusion results before and after the time-sequence transformation, and the confidence value reaches 0.6 around 40 min before the time-sequence transformation, and after the time-sequence transformation, it can reach 0.6 in 35 min. Comparing the before and after the time-sequence transformation, the signals after the time-sequence transformation are able to diagnose the system faults in 5 min in advance.

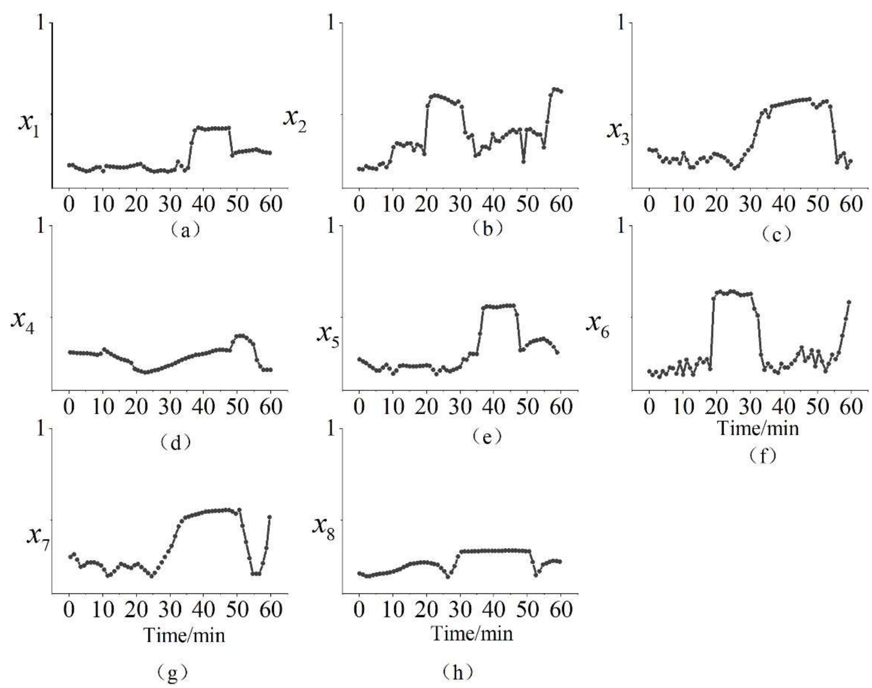


Figure 9: Reliability values of fault mode before time sequence transformation

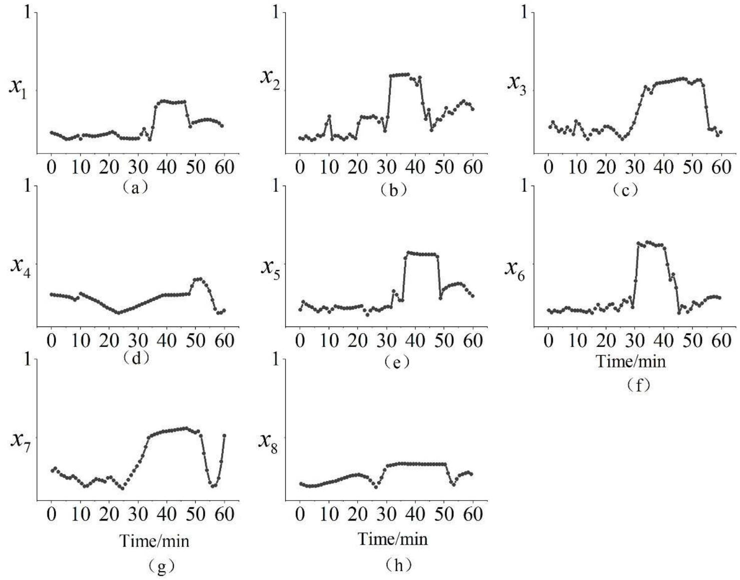


Figure 10: Reliability values of fault mode after time sequence transformation

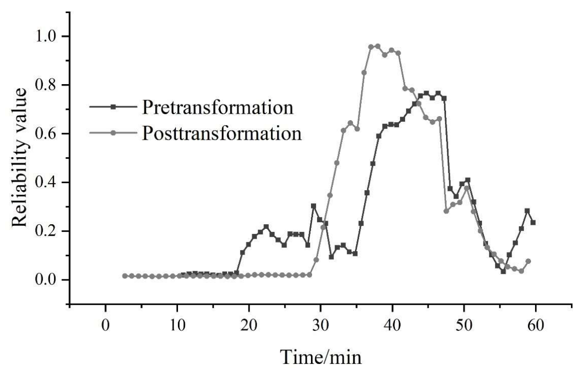


Figure 11: Failure mode reliability value before and after the sequential transformation

IV. B. Predictive maintenance

Fig. 12 shows the results of wear value prediction with time 0-180 as initial wear, time 180-490 as medium (normal) wear and 490-600 as severe wear. The system has a good fitting effect in the normal wear stage with a predicted MSE of 6.218 and a model accuracy of around 0.815 when performing wear value prediction. It is not effective in the light as well as severe wear stage, especially in the severe wear stage, and accordingly, a similar situation occurs for machine maintenance cost prediction as well as remaining life prediction.

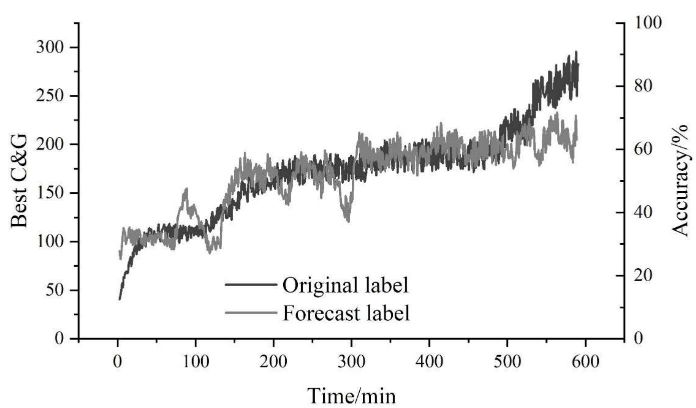


Figure 12: The wear value predicts the result

V. Conclusion

In this paper, the instantaneous Fourier transform is utilized to analyze the time-frequency variation of fault signals. The associative LSTM and CNN models are used to improve the fault classification performance of the model system, and the dynamic characteristics of the time-frequency features are captured more effectively by introducing the deep learning attention mechanism of CBAM.

The model is utilized to discriminate the type of system fault with a fault depth of 0.2746 mm, a fault diameter of 0.1758 mm, and a fault time-domain amplitude between -4 and 4. The instantaneous Fourier transform exists with high time resolution, and the highest peak value appears at 2815 Hz with an amplitude of 0.325 $m\cdot s^{-2}$ indicating that the instantaneous Fourier transform has a high time resolution.

A deep time-domain analysis model is designed to achieve predictive maintenance of system failures. The LSTM+RNN model is used to predict the production system failure state prediction, and the combined prediction recall, precision, and accuracy of LSTM+RNN are 0.964, 0.945, and 0.964, respectively. The method proposed in this paper effectively utilizes the feature extraction capability of LSTM for time series data.

The acquired raw system operation data are time series transformed, and the evidence before and after the transformation is assigned with confidence function using data fusion method respectively. Comparing before and after the time-sequence transformation, the confidence value reaches 0.6 around 40min and 35min respectively, so the signal after the time-sequence transformation is able to diagnose the system fault 5min in advance.

Funding

Funded by the training object of outstanding young backbone teachers in the Qinglan Project of colleges and universities in Jiangsu Province (2023QL004);

Enterprise practice training project for young teachers in higher vocational colleges in Jiangsu Province (2023QYSJ106);

General project of philosophy and social science research in colleges and universities in Jiangsu Province: "Research on the dynamic training mode of innovative electronic information talents adaptable to local economic development" (2024SJYB1164);

Suzhou Vocational Institute of Industrial Technology Scientific Research Startup Fund Project “Research on Fault Detection and Diagnosis Methods for Nonlinear Processes Based on Hybrid Computing” (2020kyqd002).

References

[1] Compare, M., Baraldi, P., & Zio, E. (2019). Challenges to IoT-enabled predictive maintenance for industry 4.0. IEEE Internet of things journal, 7(5), 4585-4597.

[2] Zhong, D., Xia, Z., Zhu, Y., & Duan, J. (2023). Overview of predictive maintenance based on digital twin technology. Heliyon, 9(4).

[3] Zonta, T., Da Costa, C. A., da Rosa Righi, R., de Lima, M. J., Da Trindade, E. S., & Li, G. P. (2020). Predictive maintenance in the Industry 4.0: A systematic literature review. Computers & Industrial Engineering, 150, 106889.

[4] Nunes, P., Santos, J., & Rocha, E. (2023). Challenges in predictive maintenance–A review. CIRP Journal of Manufacturing Science and Technology, 40, 53-67.

[5] Selcuk, S. (2017). Predictive maintenance, its implementation and latest trends. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 231(9), 1670-1679.

[6] Yan, W., Wang, J., Lu, S., Zhou, M., & Peng, X. (2023). A review of real-time fault diagnosis methods for industrial smart manufacturing. Processes, 11(2), 369.

[7] Zhang, W., Yang, D., & Wang, H. (2019). Data-driven methods for predictive maintenance of industrial equipment: A survey. IEEE systems journal, 13(3), 2213-2227.

[8] Serradilla, O., Zugasti, E., Rodriguez, J., & Zurutuza, U. (2022). Deep learning models for predictive maintenance: a survey, comparison, challenges and prospects. Applied Intelligence, 52(10), 10934-10964.

[9] Arena, F., Collotta, M., Luca, L., Ruggieri, M., & Termine, F. G. (2021). Predictive maintenance in the automotive sector: A literature review. Mathematical and Computational Applications, 27(1), 2.

[10] Pech, M., Vrchota, J., & Bednář, J. (2021). Predictive maintenance and intelligent sensors in smart factory. Sensors, 21(4), 1470.

[11] Tiddens, W., Braaksma, J., & Tinga, T. (2022). Exploring predictive maintenance applications in industry. Journal of quality in maintenance engineering, 28(1), 68-85.

[12] Sakib, N., & Wuest, T. (2018). Challenges and opportunities of condition-based predictive maintenance: a review. Procedia cirp, 78, 267-272.

[13] Dalzochio, J., Kunst, R., Pignaton, E., Binotto, A., Sanyal, S., Favilla, J., & Barbosa, J. (2020). Machine learning and reasoning for predictive maintenance in Industry 4.0: Current status and challenges. Computers in Industry, 123, 103298.

[14] Divya, D., Marath, B., & Santosh Kumar, M. B. (2023). Review of fault detection techniques for predictive maintenance. Journal of Quality in Maintenance Engineering, 29(2), 420-441.

[15] Yu, W., Dillon, T., Mostafa, F., Rahayu, W., & Liu, Y. (2019). A global manufacturing big data ecosystem for fault detection in predictive maintenance. IEEE Transactions on Industrial Informatics, 16(1), 183-192.

[16] Fernandes, M., Canito, A., Corchado, J. M., & Marreiros, G. (2019, June). Fault detection mechanism of a predictive maintenance system based on autoregressive integrated moving average models. In International symposium on distributed computing and artificial intelligence (pp. 171-180). Cham: Springer International Publishing.

[17] Li, J., Guo, Y., Wall, J., & West, S. (2019). Support vector machine based fault detection and diagnosis for HVAC systems. International Journal of Intelligent Systems Technologies and Applications, 18(1-2), 204-222.

[18] Mahmood, T., Li, J., Pei, Y., Akhtar, F., Butt, S. A., Ditta, A., & Qureshi, S. (2022). An intelligent fault detection approach based on reinforcement learning system in wireless sensor network. The Journal of Supercomputing, 78(3), 3646-3675.

[19] Benkercha, R., & Moulahoum, S. (2018). Fault detection and diagnosis based on C4. 5 decision tree algorithm for grid connected PV system. Solar Energy, 173, 610-634.

[20] Calabrese, F., Regattieri, A., Bortolini, M., & Galizia, F. G. (2022). Data-driven fault detection and diagnosis: Challenges and opportunities in real-world scenarios. Applied Sciences, 12(18), 9212.

[21] Xue, P., Zhou, Z., Fang, X., Chen, X., Liu, L., Liu, Y., & Liu, J. (2017). Fault detection and operation optimization in district heating substations based on data mining techniques. Applied energy, 205, 926-940.

[22] Wen, L., Li, X., Gao, L., & Zhang, Y. (2017). A new convolutional neural network-based data-driven fault diagnosis method. IEEE transactions on industrial electronics, 65(7), 5990-5998.

[23] Kiranyaz, S., Gastli, A., Ben-Brahim, L., Al-Emadi, N., & Gabbouj, M. (2018). Real-time fault detection and identification for MMC using 1-D convolutional neural networks. IEEE Transactions on Industrial Electronics, 66(11), 8760-8771.

[24] Shahnazari, H. (2020). Fault diagnosis of nonlinear systems using recurrent neural networks. Chemical Engineering Research and Design, 153, 233-245.

[25] Zhang, Y., Zhou, T., Huang, X., Cao, L., & Zhou, Q. (2021). Fault diagnosis of rotating machinery based on recurrent neural networks. Measurement, 171, 108774.

[26] Bei Li & Zhuosheng Zhang. (2024). Synchroextracting Transform Based on the Novel Short-Time Fractional Fourier Transform. Fractal and Fractional,8(12),736-736.

[27] Xianhao Jiang,Tonghai Chen,Xinhan Xiong,Jing Zeng,Yu Zhao,Ruiqin Chen... & Taihong Cheng. (2024). The self-tuning fuzzy sliding mode control method for the suspension system with the LSTM network road identification. Journal of Sound and Vibration,581,118401-.

[28] Shain Azadi,Yoji Okabe & Valter Carvelli. (2025). Bayesian-optimized 1D-CNN for delamination classification in CFRP laminates using raw ultrasonic guided waves. Composites Science and Technology,264,111101-111101.

[29] Ping Jiang,Junzi Zhang & Jiejie Chen. (2025). Enhanced rain removal network with convolutional block attention module (CBAM): a novel approach to image de-raining. EURASIP Journal on Advances in Signal Processing,2025(1),9-9.