Fault Diagnosis of Rolling Bearing Based on Convolutional Neural Network of Convolutional Block Attention Module

Chaoqun Wang1, Binbin Li1,* and Bin Jiao1
1School of Electrical engineering, Shanghai Dianji University, Shanghai, China

*Corresponding author e-mail: libb@sdju.edu.cn

Abstract. Aiming at the problem of insufficient comprehensive feature extraction in the fault diagnosis of rolling bearings, this paper proposes a fault diagnosis model of convolutional neural network (CNN) based on Convolutional Block Attention Module (CBAM). The model uses CBAM instead of pooling layer to join the CNN, and then exports to the full connected layer to realize the fault type identification and complete the fault diagnosis of rolling bearing. Experiments show that the accuracy of this method is 99.94%, and comparison with some other methods proves the effectiveness of this method.

1. Preface
Rolling bearings are one of the most important parts in mechanical equipment today. It has the reputation of "industrial joint" [1]. Because of its long-term high load operation, it's one of the most vulnerable parts in mechanical equipment. Some data indicate that rolling bearing failures account for approximately 30%-40% of motor failures [2]. Therefore, bearing fault diagnosis has always been a hot spot in domestic and foreign research. The classic bearing fault diagnosis is to extract the characteristics of the fault signal by manual, and then apply it to the fault diagnosis model. Literature [3] proposed a rolling bearing fault diagnosis method based on Hilbert marginal spectrum, which can effectively extract rolling bearing fault features. Literature [4] combined wavelet packet decomposition and sample entropy to propose a new method for rolling bearing fault feature extraction. Finally, the fault state is evaluated by sample entropy, which proves the effectiveness of the method. Literature [5] proposed a feature extraction method based on variational mode decomposition and singular value decomposition, using fuzzy c-means (FCM) for fault identification. The method is applied to the fault diagnosis of variable load of rolling bearings, and good results are obtained. In recent years, with the continuous development of artificial intelligence technology, deep learning has attracted more and more attention. It can use multiple non-linear processing layers to calculate the model, thereby automatically learning data at different levels, so it can be used to learn complex functions or complex structures in high-dimensional data [6]. At present, it has been widely used in computer vision, speech recognition, natural language processing and other fields. Through the method of deep learning, the step of extracting the fault by manual can be omitted, and vibration signal can be directly used as the input of the model to diagnose fault. Literature [7] proposed to use the hierarchical diagnosis network (HDN) by Deep Belief Network (DBN), and use hierarchical structure of network to diagnose rolling bearing faults. It was classified in bearing data set of Western Reserve University with a sampling frequency of 12 kHz, and achieved good results. Literature [8] proposed the first layer of wide convolution kernel deep convolutional neural network (WDCNN), which reduced the design difficulty of CNN. And it also achieve a high recognition rate in the bearing
data set of Western Reserve University with a sampling frequency of 12 kHz. Literature [9] uses the stacked sparse auto-encoders (SAE) to extract sparse features to diagnose faults on bearings and valves, and it has achieved a high accuracy rate on diagnose faults of bearings and valves. These models maybe have the problem of insufficient comprehensive feature extraction, but it can be seen that deep learning has great potential in bearing fault diagnosis. Based on the classic CNN model of LeNet-5 [10], this paper proposes to use CBAM [11] instead of pooling layer in each layer structure of CNN to increase feature extraction capability of the network. Finally it realizes identification of the fault type in full connected layer and completes the fault diagnosis of rolling bearing.

2. Introduction

2.1. Principle of attention mechanism
Human visual system tends to the part that helps us make judgments and ignores other irrelevant information. The essence of attention model in deep learning is the same. Not all input information is useful. We only let machine select the most useful information for the current task from many information, thereby improving computational efficiency [12].

2.2. Channel Attention Module
The function of channel attention module is to pay attention to what kinds of features are meaningful. As shown in Figure 1, after inputting an H×W×C intermediate feature F, they are respectively subjected to global max pooling and global average pooling for a space to obtain two 1×1×C channels. Then send them to the shared network separately. The shared network consists of multilayer perceptron (MLP) and a hidden layer. To reduce parameters, the size of the hidden layer is set to $R^{C/r×1×1}$. Where r is the reduction ratio. The activation function is Relu. After adding the obtained features, the weight coefficient is obtained after Sigmoid function. This process can be expressed by the following equation (1), where $\sigma$ is Sigmoid function and $W_0, W_1$ is the weight of MLP.

$$M_C(F) = \sigma(MLP(AvgPool(F)) + MLP(MaxPool(F))) = \sigma(W_1(W_0(F_{avg}^C)) + W_1(W_0(F_{max}^C)))$$  \hspace{1cm} (1)

![Figure 1. Channel Attention Module](image)

2.3. Spatial Attention Module
The role of spatial attention module is to focus on where the features are meaningful. As shown in Figure 2, similar to channel attention module, after inputting an H×W×C intermediate feature F, they are subjected to max pooling and average pooling of one channel to obtain two H×W×1 aisle. After concatenating them together, they are sent to a convolutional layer with a convolution kernel of size 7×7, activation function is Sigmoid, and finally the weight coefficient $M_S(F) \in R^{H×W}$ is obtained. This process can be expressed by the following equation (2), where $\sigma$ is Sigmoid function and $f^{7×7}$ means that the size of the convolution kernel is 7×7.

$$M_S(F) = \sigma(f^{7×7}([AvgPool(F); MaxPool(F)])) = \sigma(f^{7×7}([F_{avg}^S; F_{max}^S]))$$  \hspace{1cm} (2)
2.4. Convolutional Block Attention Module

According to experimental results, the best way for CBAM is to connect two modules in series, and the channel attention module is placed before the spatial attention module [11]. As shown in Figure 3[11], given an intermediate feature $F$, the module infers the attention map in sequence along two independent dimensions (channel and space), and then multiplies the weight by the input feature $F$ to refine the feature. This process can be expressed by the following equation (3), where $\otimes$ means element-wise multiplication. Since CBAM is a lightweight general module, it can be seamlessly integrated into any CNN architecture. Its overhead is negligible, and it can be trained "end-to-end" with CNN.

$$F' = M_C(F) \otimes F, F'' = M_S(F') \otimes F'$$

3. Design of fault diagnosis model

3.1. CBAM

First, the input is subjected with max pooling, and then the feature passes through a full connected layer. In order to reduce parameters, the output nodes here is $\text{channel} // \text{ratio}$, and the next step is go through a full connected layer again, so that the output nodes is channel. "channel" is the number of convolution kernels of the previous convolutional layer, "ratio"is the reduction ratio $r$, and it's set to 16 in this experiment, "//"indicates division, and activation function is Relu. At the same time, the input is pooled with average pooling, and then the operation is the same as max pooling. Then add two features, and the result weight coefficient $M_S$ is obtained through the output of Sigmoid function. And the final result is multiplied with the input result to obtain the feature $X$. After two features obtained, they are subjected with a max pooling and an average pooling, the obtained results are connected with the channel of Axis = 3. Then let them pass through a convolution layer, and activation function is Sigmoid. Finally, the result weight coefficient $M_C$ and $X$ are multiplied to obtain the final output.

3.2. Fault diagnosis model in this paper

According to the classic models of CNN: LeNet-5 and CBAM, the rolling bearing fault diagnosis model of this paper is constructed. As shown in Table 1, the input is bearing vibration signal, the first layer is a convolution layer, the size of the convolution kernel is $5 \times 5$, the depth of the convolution
kernel is 64, the strides is 1. And the second layer is CBAM. The third layer is a convolution layer, the size of the convolution kernel is 3×3, the depth of the convolution kernel is 32, the strides is 1. The fourth layer is CBAM, and the parameter settings are the same as the second layer. The fifth layer is a full connected layer with 512 output nodes. The sixth layer is Softmax layer. Through this layer, the output values of multiple classifications are converted into relative probabilities, and finally the identification of the rolling bearing fault is realized [13].

Table 1. Network structure

| Number | Network layer | Size    | Strides | Depth | Padding |
|--------|---------------|---------|---------|-------|---------|
| 0      | Input         | 20×20×1 | -       | -     | -       |
| 1      | Conv-1        | 5×5     | 1       | 64    | Same    |
| 2      | CBAM          | -       | -       | -     | -       |
| 3      | Conv-2        | 3×3     | 1       | 32    | Same    |
| 4      | CBAM-2        | -       | -       | -     | -       |
| 5      | Dense         | 512     | -       | 1     | Valid   |
| 6      | Softmax       | 10      | -       | 1     | Valid   |

4. Experiment

4.1. Data preparation

The data used in this experiment comes from the rolling bearing data set of Western Reserve University. Select the drive end bearing of model SKF-6205, the load is 2 horsepower, the speed is 1750RPM, and the signal sampling frequency is 12kHz driver data. It is divided into four states: normal state, rolling element fault, inner ring fault and outer ring fault. The three fault states are divided into three status with a fault diameter of 0.007, 0.014, and 0.021 inch. These faults are all artificially processed by EDM. I used acceleration data of driver end (DE) for the experiment. The specific fault specifications are shown in Table 2 below.

Table 2. Data set

| Label | Fault location | Fault size(in) | Code name | Training sample | Validation sample | Test sample |
|-------|----------------|----------------|-----------|-----------------|-------------------|-------------|
| 0     | Normal         | 0              | Normal    | 3500            | 1000              | 500         |
| 1     | Rolling element| 0.007          | B007      | 3500            | 1000              | 500         |
| 2     |                | 0.014          | B014      | 3500            | 1000              | 500         |
| 3     |                | 0.021          | B021      | 3500            | 1000              | 500         |
| 4     | Inner ring     | 0.007          | IR007     | 3500            | 1000              | 500         |
| 5     |                | 0.014          | IR014     | 3500            | 1000              | 500         |
| 6     |                | 0.021          | IR021     | 3500            | 1000              | 500         |
| 7     | Outer ring@6   | 0.007          | OR007     | 3500            | 1000              | 500         |
| 8     |                | 0.014          | OR014     | 3500            | 1000              | 500         |
| 9     |                | 0.021          | OR021     | 3500            | 1000              | 500         |

Since there are ten states in total, the ten states are set to label 0-9, representing normal signals and 9 fault signals. For the convenience of input, the sampling method of continuous sampling is used to obtain samples, and the sampling step is set to 400 [14]. Randomly extract 5000 samples from each signal, add a label and finally get a 50000×(400 + 1) matrix, save this matrix to a .csv file as the network input. During model training, the data is divided into a training set, a validation set and a test set in a 7: 2: 1 manner.
4.2. Parameter setting

This experiment was completed in Google's deep learning framework TensorFlow. The experiment set the batch size to 128, and the Epoch is 12. Set activation function to Relu in the convolution layer. Set L2 regularization coefficient to $10^{-4}$, and set activation function to Relu in the full connected layer. Then flatten the feature matrix and put it into the full connected layer. Before put it into the Softmax layer, add a Dropout layer. The dropout rate is 0.5. Set activation function to softmax in the Softmax layer, L2 regularization coefficient is $10^{-4}$, and L2 regularization coefficient of the output term is $10^{-3}$. Select Adam optimizer, the optimization coefficient is $10^{-4}$, and adopt the dynamic attenuation method, the decay rate is $10^{-8}$. The loss function selects cross-entropy loss function.

4.3. Experimental results

As shown in Figures 4 and 5, after 12 Epochs, the final accuracy rate on the test set is 99.94%, and the loss is about 0.038.

![Figure 4. Accuracy on the test set](image1)

![Figure 5. Loss values on the test set](image2)

Figure 6 shows the confusion matrix corresponding to the test set. It can be seen from the figure that except for B007 with a misdiagnosis rate of 0.39% and B021 with a misdiagnosis rate of 0.21%, the diagnosis is basically correct in other cases.

Evaluating the quality of a bearing fault diagnosis model is not only the accuracy rate, but also precision, recall rate, F1 Score and other indicators [15]. Accuracy and F1 Score are usually used to measure the overall performance of the model. The higher the index, the stronger the diagnostic capability of the model and the better the overall performance. As shown in Table 3, precision, recall rate, and F1 Score of the model are all 1.00, which fully shows that the overall performance of the model is great.
4.4. Comparison with some other algorithms

In order to prove the effectiveness of the model, this paper uses Naive Bayes Model (NBM), K-Nearest Neighbor (KNN), Support Vector Machine (SVM), DNN and CNN models for comparative experiments. Use the same 35,000 sets of data as section 3.1 as input data, 10,000 sets of data as the validation set, and 5000 sets of data as the test set. Among them, the Gaussian Bayesian classifier is used in the Naive Bayes Model; K=2 is set in the K-nearest Neighbor; the kernel function in the Support Vector Machine is set to Gaussian kernel function, parameter C=1, γ=0.1; the network structure in DNN is 1024-512-10; the network structure in CNN is 64×64×5-32×32×3-512-10. The results are shown in Table 4 below. It can be seen from the table that the accuracy of the model proposed in this paper is higher than these models, which proves the effectiveness of the model.

**Table 4.** Comparison with some other algorithms

| Diagnostic model | Accuracy / % |
|------------------|--------------|
| NBM              | 58.09        |
| KNN              | 91.48        |
| SVM              | 97.54        |
| DNN              | 98.84        |
| CNN              | 99.76        |
5. Conclusion

Based on the classic CNN model of LeNet-5, this paper proposes to use CBAM instead of the pooling layer in each layer of CNN as a fault diagnosis method and implement it in the deep learning framework TensorFlow. Based on the powerful processing capabilities of deep learning, high-precision rolling bearing fault diagnosis can be achieved without manual feature extraction. And in the experiment finally reached 99.94% accuracy. And compared with some other algorithms, the effectiveness of this method is proved.

References

[1] Gang Deng. Research on fault diagnosis method of rolling bearing based on deep learning [D]. Dalian University of technology, 2019:1.
[2] Hongzhong Ma. Motor condition monitoring and fault diagnosis [M]. Beijing: Machinery Industry Press, 2008:676.
[3] Yu Yang, Dejie Yu, Junsheng Cheng. Fault diagnosis method of rolling bearing based on Hilbert marginal spectrum [J]. Vibration and Shock, 2005(01): 72-74 + 137.
[4] Wensheng Su, Fengtao Wang, Hong Zhu, Zhenggang Guo, Zhixin Zhang, Hongyin Zhang. Feature extraction of rolling bearing faults based on wavelet packet sample entropy [J]. Vibration: Testing and Diagnostics, 2011, 31(02): 162-166 + 263.
[5] Changliang Liu, Yingjie Wu, Chenggang Zhen. Fault diagnosis of rolling bearings based on variational modal decomposition and fuzzy C-means clustering [J]. Chinese Journal of Electrical Engineering, 2015, 35(13): 3358-3365.
[6] LeCun Yann, Bengio Yoshua, Hinton Geoffrey. Deep learning [J]. Nature, 2015, 521(7553).
[7] Gan M, Wang C. Construction of hierarchical diagnosis network based on deep learning and its application in the fault pattern recognition of rolling element bearings [J]. Mechanical Systems and Signal Processing, 2016, 72: 92-104.
[8] Zhang W, Peng G, Li C, et al. A New Deep Learning Model for Fault Diagnosis with Good Anti-Noise and Domain Adaptation Ability on Raw Vibration Signals [J]. Sensors, 2017, 17(2): 425.
[9] Verma NK, Gupta VK, Sharma M, et al. Intelligent condition based monitoring of rotating machines using sparse auto-encoders [C] // Prognostics and Health Management (PHM), 2013 IEEE Conference on. IEEE, 2013: 1-7.
[10] Lecun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition [J]. Proceedings of the IEEE, 1998.
[11] Woo S, Park J, Lee J Y, et al. CBAM: Convolutional Block Attention Module [J]. 2018.
[12] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need [J]. Neural Information Processing Systems, 2017: 5998-6008.
[13] Zeyu Zheng, Bowen Liang, Siyu Gu. TensorFlow actual combat Google deep learning framework [M]. Electronic Industry Press, 2018: 75-76.
[14] Yu Liang, Binbin Li, Zhigao Chen, Bin Jiao. Research on deep feedforward network optimization in TensorFlow and its application in bearing fault diagnosis [J]. Computer Applications and Software, 2019, 36(10): 175-182.
[15] Sokolova M, Lapalme G. A systematic analysis of performance measures for classification tasks [J]. Information Processing and Management, 2009, 45(4): 427-437.