 doi:10.20944/preprints202008.0164.v1

Article

An anti-noise fault diagnosis method of bearing based on multi-scale 1DCNN

Jie Cao [1, 2, 3], Zhidong He[1,*],Jinhua Wang [1] and Ping Yu[1]

[1] College of Electrical and Information Engineering, Lanzhou University of Technology, Lanzhou 730050, China; caoj@lut.edu.cn (J.C.); osunmmo@foxmail.com (Z.H.); wjh0615@lut.edu.cn(J.W.); 25464714@qq.com(P.Y.).
[2] Engineering Research Center of Urban Railway Transportation of Gansu Province, Lanzhou 730050, China;
[3] Engineering Research Center of Manufacturing Information of Gansu Province, Lanzhou 730050, China;
* Correspondence: osunmmo@foxmail.com;

Abstract: In recent years, intelligent fault diagnosis algorithms using deep learning method have achieved much success. However, the signals collected by sensors contain a lot of noise, which will have a great impact on the accuracy of the diagnostic model. To address this problem, we propose a one-dimensional convolutional neural network with multi-scale kernels (MSK-1DCNN) and apply this method to bearing fault diagnosis. We use a multi-scale convolution structure to extract different fault features in the original signal, and use the ELU activation function instead of the ReLU function in the multi-scale convolution structure to improve the anti-noise ability of MSK-1DCNN; then we use the training set with pepper noise to train the network to suppress overfitting. We use the Western Reserve University bearing data to verify the effectiveness of the algorithm and compare it with other fault diagnosis algorithms. Experimental results show that the improvements we proposed have effectively improved the diagnosis performers of MSK-1DCNN under strong noise and the diagnosis accuracy is higher than other comparison algorithms.

Keywords: intelligent fault diagnosis; bearing; anti-noise; one-dimensional convolution neural network

1. Introduction

Rolling bearings are not only an essential component, but also the main factor leading to system failures in rotating machinery. 45%-55% of equipment failures are caused by bearing damage [1]. Every unexpected failure of the bearing may lead to the failure of machine and even entire system, and result in huge economic losses and a waste of time. As a major problem in the field of fault diagnosis, the fault diagnosis of rolling bearings has attracted extensive attention from researchers. The traditional method of bearing fault diagnosis is to analyze the vibration signal collected by the sensor, and then use the intelligent algorithm to extract the fault characteristics of the signal, and finally use the classification algorithm to detect fault type. With the rapidly rising of deep learning and its successful applications in computer vision [2], natural language processing [3], medical image analysis [4]and other fields, intelligent fault diagnosis algorithms based on deep learning have also been rapid development in recent years [5,6]. Deep learning algorithms for bearing fault diagnosis mainly include Autoencoders (AE), Restricted Boltzmann Machines (RBM) and Convolutional Neural Networks (CNN), etc.

Compared with AE and RBM, convolutional neural networks have advantages in processing time series data and vibration signals with variable translation characteristics [7], so researchers have used one-dimensional convolutional neural networks to directly extract fault features from original signal to classify faults in recent years. T Ince et al. [8] use a one-dimensional convolution neural network to process the current signal of the motor. The proposed one-dimensional convolution network is very effective in calculation and can be easily and cheaply implemented on hardware systems. Levent Eren [9] uses a one-dimensional convolutional neural network to quickly and

 doi:10.20944/preprints202008.0164.v1

accurately detect motor bearing faults, with an accuracy rate of 97.1%. Wei Zhang et al. [10] proposed a deep convolutional neural network with a large first-layer convolution kernel (WDCNN). The proposed method uses original vibration signal as input, uses a large convolution kernel in the first convolution layer to extract features and suppress high-frequency noise, and then uses small convolution kernels in next layers of the network to achieve multi-layer nonlinearity mapping, while using AdaBN to improve the domain adaptability of the model. In another paper, Wei Zhang et al. [6] proposed a method called TICNN (Convolution Neural Network with Training Interference) for the problem of a lot of noise and variable operating conditions in the working environment of the bearing, which directly extracts the fault characteristics from the original vibration signal without additional data preprocessing. TICNN has made the following improvements: 1) Convolution kernel dropout is used in the first convolutional layer, 2) Small batch training is used in the optimization algorithm and ensemble learning is used to improve the stability of the network.

The current one-dimensional fault diagnosis model basically achieves a 100% fault recognition rate under no-noise conditions, showing the powerful feature extraction ability of convolutional neural networks. However, most of the currently proposed models do not consider the situation that the signal contains noise. The signal collected by sensors in the real working environment contains a lot of noise, which will have a great impact on the accuracy of diagnostic model. Therefore, most models have not achieved good diagnostic accuracy in the presence of noise. To address this problem, we propose a one-dimensional convolutional neural network with multi-scale convolution kernels (MSK-1DCNN). MSK-1DCNN directly acts on original vibration signal, and the feature extraction and fault classification are realized through the convolutional neural network. MSK-1DCNN has made the following improvements: 1)Pepper noise is added to the input training data during the network training stage to increase the complexity of the input signal and improve the network's ability to extract generalized features; 2)At the front of the network, a single-layer and single-kernel convolution layer is replaced with two layers multi-scale convolution structure to extract more diverse fault features; 3) In the multi-scale convolution structure, the ELU activation function is used instead of the Relu function to improve the anti-noise ability of the network.

The remainder of this paper is organized as follows: in Section 2, a introduction of CNN is presented. The proposed MSK-1DCNN model is introduced in Section 3. Section 4 presents and discusses the result from different experimental conditions. A comparison is also made with proposed method. We draw the conclusions in Section 5.

2.Introduction of Convolutional Neural Networks

The convolutional neural network is a multi-level feedforward neural network, which is usually composed of three types of layers: convolutional layer, pooling layer and fully connected layer. The convolutional layer and the pooling layer extract the characteristics of input data through convolution calculation and down-sampling operations, and then the fully connected layer achieves classification or regression task. The fully connected layer has the same structure and calculation method as the traditional feedforward neural network.

1.1 Convolutional layer

The convolutional layer learns the features of input data through convolution calculation. It is composed of multiple feature maps. Each neuron of each feature map is connected to a local area of the previous layer of feature maps through a set of weights. This local area is called the receptive field of the neuron, and this set of weights is called the convolution kernel. By performing convolution calculation on the input feature map and the convolution kernel, and then transferring the result to the nonlinear activation function, the next layer feature map is generated. The convolutional layer uses different convolution kernels to generate different feature maps. A single feature map is calculated by the same convolution kernel, which is called weight sharing. Weight sharing can reduce

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 6 August 2020 doi:10.20944/preprints202008.0164.v1

the complexity of the model and make the network easier to train. The forward propagation of the convolutional neural network from layer $l-1$ to layer l can be expressed by the following formula [11]:

$$x_j^l = f\left(\sum_{i \in M_j} x_i^{l-1} \cdot k_{ij}^l + b_j^l\right) \quad (1)$$

where x_j^l represents the output of the layer l, M_j represents the selected feature map, x_i^{l-1} represents the output of the layer $l-1$, k_{ij}^l represents the weight of layer l, and b_j^l represents the bias of layer l.

1.2 Activation layer

The activation function is usually used to implement a non-linear transformation on the output of convolution calculation to obtain a non-linear representation of input data, thereby improving the feature learning ability of the network. The activation function commonly used in CNN is the Rectified Linear Unit (ReLU) function, and its calculation formula is [12]:

$$f(x)\, max(0, x)_{cov} \quad (2)$$

Where x is the input of activation function.

In order to improve the anti-noise ability of the model, we use Exponential Linear Unit (ELU) activation function in the multi-scale convolution structure, which can speed up the learning process and improve the accuracy of the network. Similar to ReLU function, ELU avoids the problem of gradient disappearance by setting the positive part of the input to be identical. But unlike ReLU, ELU does not set the negative value to zero, which is beneficial to speed up the learning speed of the network. And it uses a saturation function in the negative part to make ELU more robust to noise [13]. Its calculation formula is [13]:

$$y_i = \begin{cases} z_i & z_i \geq 0 \\ a(exp(z_i) - 1) & z_i < 0 \end{cases} \quad (3)$$

where y_i is the activation function output value, z_i is the activation function input value, and a is a predefined parameter used to control the saturation value of ELU for the negative input.

1.3 Pooling layer

In the structure of convolutional neural networks, pooling layers are usually inserted between successive convolutional layers. Its role is to gradually reduce the dimension of the output of convolutional layer to reduce the parameters and calculations in the network, and also suppress overfitting and implement secondary feature extraction. The pooling layer is composed of multiple feature maps, and its feature maps correspond to the feature maps of the previous convolutional layer one by one without changing the number. The most commonly used pooling methods are maximum pooling and mean pooling. In this paper, the maximum pooling method is used because the performance of maximum pooling in one-dimensional time series tasks is better than average pooling [14]. Its calculation formula is [10]:

$$P_i^{l+1}(j) = \max_{(j-1)W+1 \leq t \leq jW} \{q_i^l(t)\} \quad (4)$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 6 August 2020 doi:10.20944/preprints202008.0164.v1

where $q_i^l(t)$ represents the output of the tth neuron in the ith feature map of the layer l, $t \in [(j-1)W+1, jW]$, W is the width of the pooled area, and $P_i^{l+1}(j)$ is the pooled value of the corresponding neuron in the layer $l+1$.

3. Proposed MSK-1DCNN Model

3.1 Multi-scale convolution structure

For the time series classification tasks using a one-dimensional convolutional neural network, the size of convolution kernel has a huge impact on the performance of the network, because part of the noise in time series cannot be removed by BN, Bias, ReLU and other operators, and it can only be eliminated by convolution operation of convolution kernel [15]. The traditional one-dimensional convolutional neural network treats the size of the convolution kernel as a hyperparameter, and uses a fixed-size convolution kernel in convolution layer, which makes the design of convolution kernel size a very difficult problem. And the use of this method in prediction and classification tasks is limited because of the following problems: 1) Large-scale convolution kernels tend to focus on low-frequency regions and have good frequency resolution, but there are not enough convolution kernels in high-frequency regions, thereby ignoring high-frequency information. In contrast, small-scale convolution kernel focuses on the frequency band, but the frequency resolution is low. 2) Using convolution kernels of the same size cannot adequately extract different discriminant features in original signal [16]. In order to solve the above problems, scholars have proposed multi-scale convolution. Multi-scale convolution uses multiple filter banks of different scales to extract features from the original signal, and has been successfully applied in many fields, such as Environmental Sound Classification [17], Speech Recognition [18], and so on. Inspired by their work, we designed a two layers multi-scale kernel feature extraction structure (MSK), as shown in Figure 1.

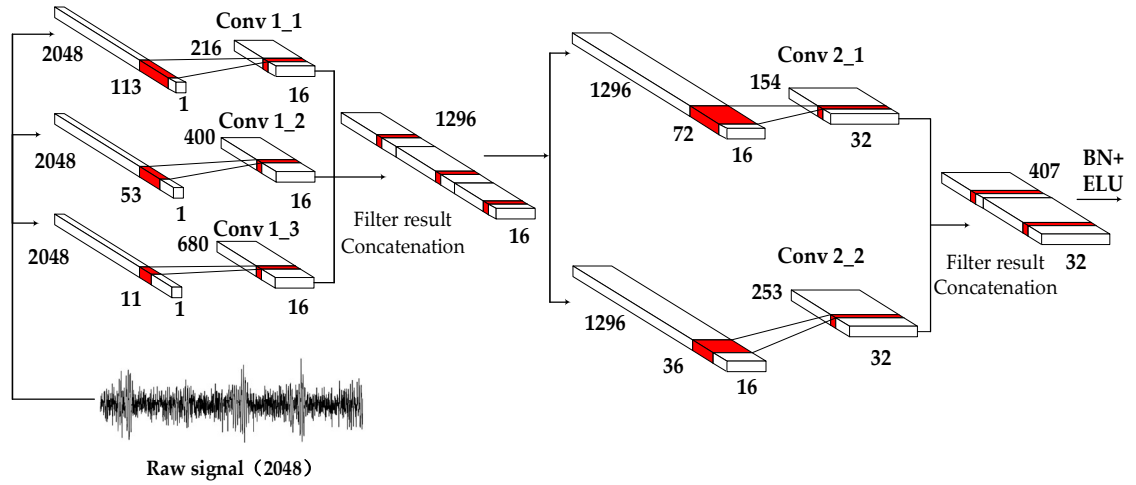


Figure 1. Multi-scale convolutional structure

In the first layer, MSK uses convolution kernels with a width of 11,53 and 113 (the number is 16) to extract features from the original data to obtain three different feature maps, and then stitch the three feature maps together as the output of the first layer. The second layer uses convolution kernels with a width of 36 and 72 (the number is 32) to continue extracting features from the output of the first layer, and then the convolution calculation results are stitched together, and finally through the BN and ELU activation function layer to get the output feature map of MSK.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 6 August 2020 doi:10.20944/preprints202008.0164.v1

3.2 Network structure and parameters of MSK-1DCNN

In addition to the multi-scale convolution structure, the proposed MSK-1DCNN has made the following improvements to the anti-noise problem.

(1) The BN layer is added after the convolution layer. The BN layer is usually used before the activation function of the convolutional layer to readjust the data distribution. Its implementation steps are described in Algorithm 1[19]. In a sense, γ and β represent the variance and offset of the input data distribution. For a network without BN, these two values are related to the nonlinear properties brought by previous layer of the network. After transformation, it is not related to previous layer and becomes a learning parameter of current layer, which is more conducive to optimization and does not reduce the ability of network [20]. BN can not only reduce the offset of covariance within input data and speed up the training process of deep neural networks, but also reduce the dependence of network on parameter initialization and increase the generalization ability.

Algorithm 1 BN

Input: value of x over a mini-batch: B = {x_1... m}; Parameters to be learned: γ, β

Output: { $y_i = BN_{\gamma,\beta}(x_i)$ }

$\mu_B = \frac{1}{m}\sum_{i=1}^{m} x_i$ //mini-batch mean

${\sigma_B}^2 = \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_B)^2$ // mini-batch variance

$\hat{x_i} = \frac{x_i - \mu_B}{\sqrt{{\sigma_B}^2 + \varepsilon}}$ //normalize

$y_i = \gamma\hat{x_i} + \beta = BN_{\gamma,\beta}(x_i)$ //scale and shift

(2) The ELU activation function is used instead of the ReLU function in MSK structure. ELU function avoids the problem of gradient disappearance and is more robust to noise, which can improve the anti-noise ability of network.

The structure and parameters of MSK-1DCNN are shown in Figure 2. The input of network is a standardized fault vibration timing signal. The network directly extracts features from original signal without any other signal processing. MSK-1DCNN model consists of a feature extraction layer and a classification layer. The front of feature extraction layer is a two-layer multi-scale convolution structure, followed by a three-layer single convolution kernel convolution layer. The multi-scale convolution structure can extract different discriminative features from original signal, and the single-convolution kernel convolution layer can extract more advanced features and deepen the depth of network to improve the anti-noise ability of model. The classification layer is composed of two fully connected layers and a softmax layer. The softmax function is used to convert network output into a probability distribution form that conforms to ten fault states of the bearing. The formula of softmax is as follows:

$$q(z_j) = \frac{e^{z_j}}{\sum k^{10} e^{z_k}} \tag{5}$$

where z_j represents the normalized probability of output of jth neuron through softmax function.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 6 August 2020 doi:10.20944/preprints202008.0164.v1

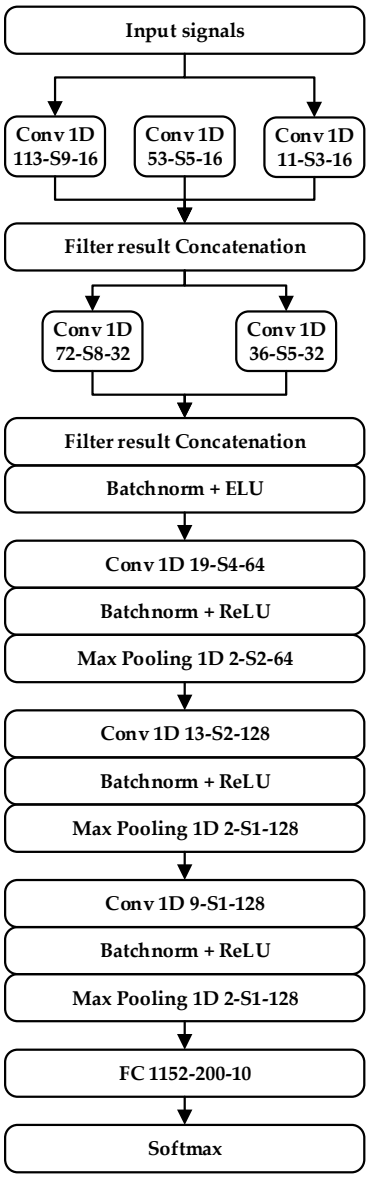


Figure 2. Architecture and parameters of MSK-1DCNN

3.3 MSK-1DCNN fault diagnosis model

MSK-1DCNN fault diagnosis model is established in three steps:

（1）Data preprocessing. The original data is cut into a data set according to the resampling method [21]. The data set is divided into a training set and a test set according to a certain ratio, and then pepper noise is added to training set, and Gaussian white noise (simulating noise in daily industrial production) is added to testing set.

（2）Training model. We select Adam optimizer and cross-entropy loss function. The Adam algorithm is easy to implement, and has high computational efficiency with low memory requirements [22]. Its optimization performance is better than SDG and RMSprop optimizer. The cross-entropy function was chosen because it is an entropy-shaped loss function, which is insensitive to noise and suitable for strong noise environments [23]. The initial value of the learning rate is set to 0.0005 and decreases by 0.0001 every 10 iterations. It does not decrease until the learning rate is 0.0001, and is fixed at 0.0001. The model is trained on the training set until the loss value is fully converged, and then stop the iteration and save the trained model.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 6 August 2020 doi:10.20944/preprints202008.0164.v1

（3）Testing model. We use the testing set to test the model and take the average of five testing results as the final testing result.

4. Experiment

4.1 Data set

The bearing data set is provided by Western Reserve University Bearing Data Center (https://csegroups.case.edu/bearingdatacenter), and its fault test bench is shown in Figure 3.

We selected the motor drive end bearing data sampled at 48kHz at 1hp, 2hp, and 3hp. The fault type of data includes normal, inner ring failure, outer ring failure and roller failure. Each fault contains 3 fault levels with widths of 0.007, 0.014, and 0.021 inches, so the data set has ten states. According to the resampling method [22], the original data of ten states are divided. Each sample has a length of 2048 and a sampling step size of 480. Therefore, each state obtains 1000 samples at 1hp, 2 hp, and 3 hp respectively, and a total of 30,000 samples in ten states of three loads constitute a data set. We divide 85% of data set into a training set and add pepper noise, and 15% of data set are divided into a test set and added with Gaussian noise of different SNR. SNR is the signal noise ratio, which represents the ratio of the power of original signal to noise power, usually expressed in decibels. The smaller the value of SNR, the stronger the noise. The formula of SNR is as follows:

$$SNR_{\mathrm{dB}}=10\log_{10}\left(\frac{P_{\mathrm{signal}}}{P_{\mathrm{noise}}}\right) \tag{6}$$

where P_{signal} and P_{noise} are the effective power of signal and noise respectively.

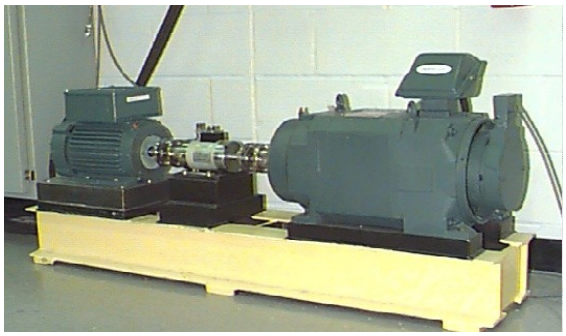


Figure 3. Motor driving mechanical system used by CWRU

4.2 Effectiveness of multi-scale convolution structure

In order to verify the effectiveness of proposed multi-scale convolution structure, we compared MSK-1DCNN with a network using a single layer with a single convolution kernel. The first layer of the single-convolution kernel network uses five convolution kernels (widths of 11, 36, 53, 72, and 113, respectively) in MSK shown in Figure 1 to extract features from original signal. The network structure after that is consistent with the network structure after multi-scale convolution structure of MSK-1DCNN. The specific parameters of single kernel convolution layer are shown in Table 1. In order to maintain the consistency of the rest of network structure, we did padding for input data.

The experimental results are shown in Figure 4. It can be seen that compared with the single-layer and single-size convolution kernel, the accuracy of network using a multi-scale convolution structure at low SNR is significantly improved. This shows that the multi-scale convolution structure

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 6 August 2020 doi:10.20944/preprints202008.0164.v1

takes into account both high-frequency region and low-frequency region, and can extract more diverse discriminative features from original signal and is more robust to noise. At the same time, it can also be found that the diagnostic accuracy of single convolution kernel networks of various sizes is obviously different, indicating that the size of convolution kernel will have a significant impact on the network diagnostic accuracy, and also reflecting the importance of convolution kernel design.

Table 1. Parameters of single-scale convolutional structure

Kernel Size	Stride	Kernel Number	Padding
11	5	32	0
36	5	32	9
53	5	32	18
72	5	32	27
113	5	32	48

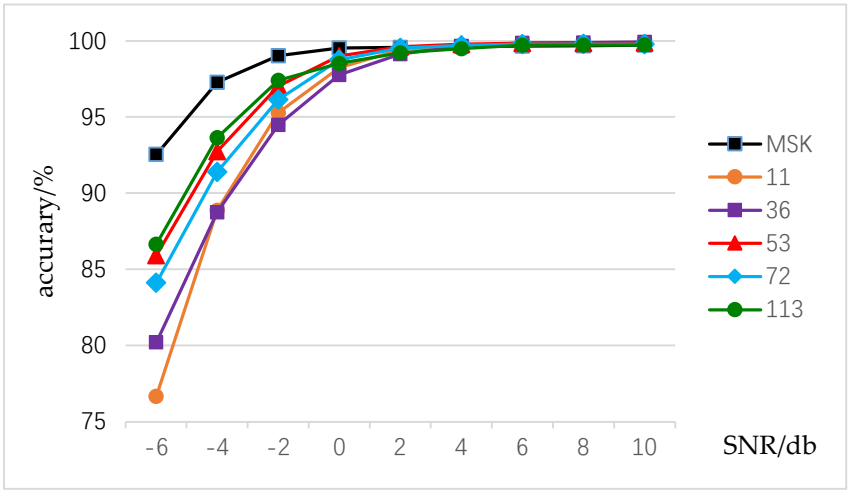


Figure 4. Diagnosis accuracy of multi-scale CNN and single CNN model

4.3 Effect of pepper noise on the performance of network feature extraction

Salt and pepper noise, also known as impulse noise, is a white and dark spot noise generated by the image sensor, transmission channel, and decoding processing in the image. Pepper is a black point (pixel value 0) and salt is a white point (pixel value 225). Generally, pepper and salt noise are added by randomly changing some pixel values to 0 or 225. When training a Denoising Auto Encoder (DAE), pepper noise is usually added to the input training data to improve the feature extraction capability of the autoencoder. This method is realized by randomly setting the input data to 0 with a certain probability, that is, dropout the input data with a certain probability [24]. Because of the successful application of this method in DAE, we add pepper noise to training data through randomly zeroing it with a probability of 0.5.

We compared the effects of adding pepper noise and not adding pepper noise to training data on network feature extraction ability (the other structures and parameters remain the same) through experiments. The results are shown in Table 2. It can be seen that the accuracy of network with noisy training is higher than that without noise when the SNR is low. It shows that adding pepper noise in

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 6 August 2020 doi:10.20944/preprints202008.0164.v1

training set can effectively destroy the original data, make the network learn more essential features of input data, suppress overfitting and improve accuracy of network under noise.

Table 2. Diagnosis accuracy of noise and no-noise input training models

SNR/db	-6	-4	-2	0	2	4	6	8	10
With noise	92.55%	97.27%	99.03%	99.53%	99.58%	99.64%	99.60%	99.68%	99.72%
With no noise	61.81%	78.17%	89.01%	95.43%	98.17%	99.19%	99.64%	99.79%	99.87%

4.4 Activation function

ReLU is the most widely used activation function in neural networks. Its simple operation of taking the maximum value makes its calculation speed much faster than Sigmoid or tanh activation function. And it also caused the sparsity in the hidden units and there was no problem of gradient disappearance. But its operation of zeroing negative values will cause the death of neurons and is not robust to noise. ELU function retains the advantages of ReLU function, and instead of zeroing the negative value part, the saturation function is used in negative value part, which makes the ELU more robust to noise.

We use the ELU activation function in the MSK, and compare it with MSK using ReLU through experiments. Except for the different activation functions of two networks, the other structures and parameters remain the same. The results are shown in Figure 5. It can be found that ELU function performs better than ReLU function at low SNR. It is proved that ELU activation function is insensitive to noise and has strong anti-noise ability under a strong noise environment. Therefore, the MSK-1DCNN using ELU function achieves better diagnostic performance at low SNR.

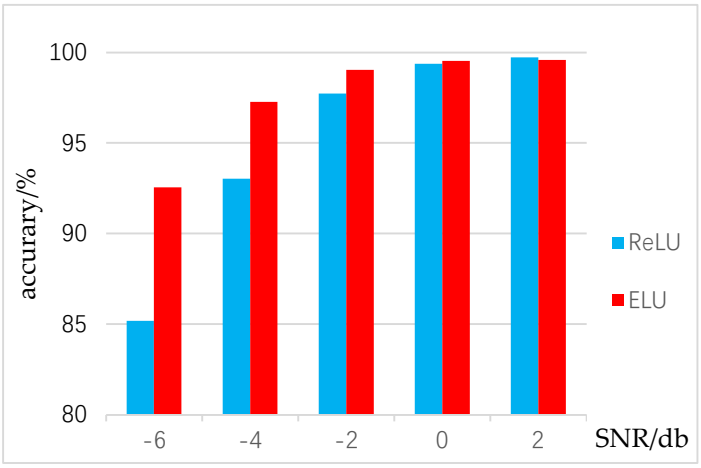


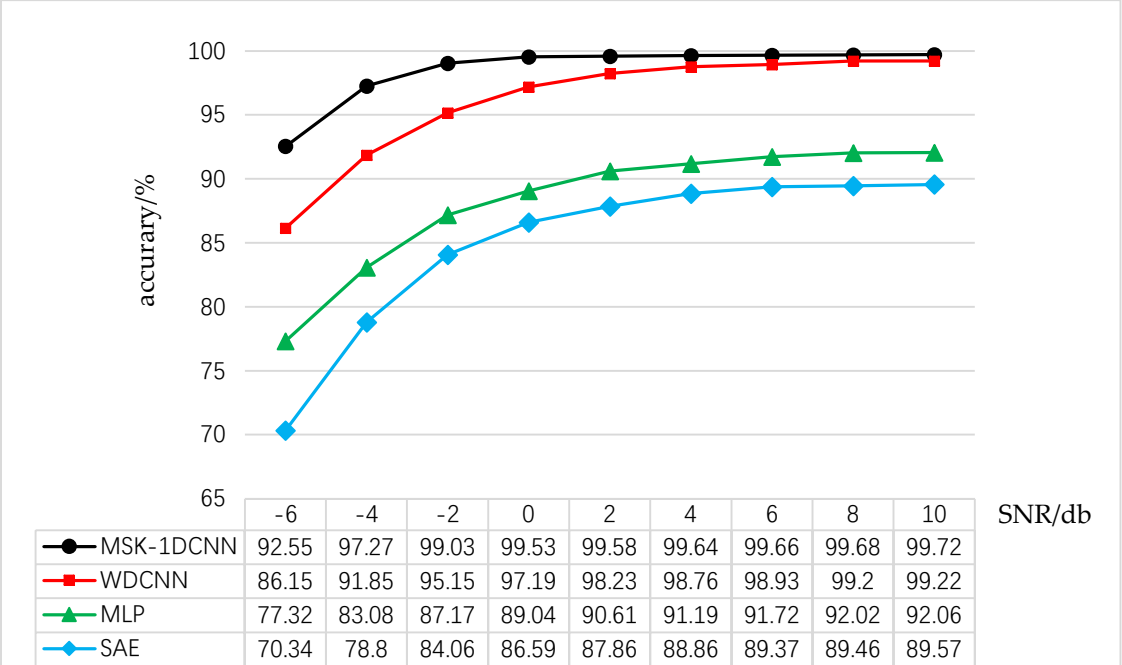
Figure 5. Diagnosis accuracy of MSK-1DCNN with ReLU or ELU

4.5 Comparison of fault diagnosis accuracy

In order to verify the effectiveness of proposed MSK-1DCNN fault diagnosis model, we select WDCNN model proposed by Wei Zhang et al. [10], Stacked Auto Encoder (SAE) and BP neural network as comparative models. We chose the WDCCNN model for comparison because its accuracy at low SNR is higher than other one-dimensional convolution models proposed in recent years. SAE is formed by stacking three autoencoders, and the number of neurons in the middle layer is 800, 200,

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 6 August 2020 doi:10.20944/preprints202008.0164.v1

and 50, respectively. The number of neurons in BP neural network is 2048, 1000, 500, 200, 10, and the Sigmoid activation function and cross-entropy loss function are used. It can be seen from figure 6 that the diagnostic accuracy of the convolutional structure is significantly higher than SAE and BP neural network, because the convolutional structure has more advantages in processing one-dimensional time series data and has stronger noise resistance. The full connection structure of BP neural network and SAE leads to serious network overfitting, so even if the SNR is high, the diagnostic accuracy of them is low. The diagnosis accuracy of the MSK-1DCNN fault diagnosis model we proposed at low SNR is significantly higher than other models, which proves the effectiveness of improvements made in this paper of bearing faults diagnosis under the noise environment.

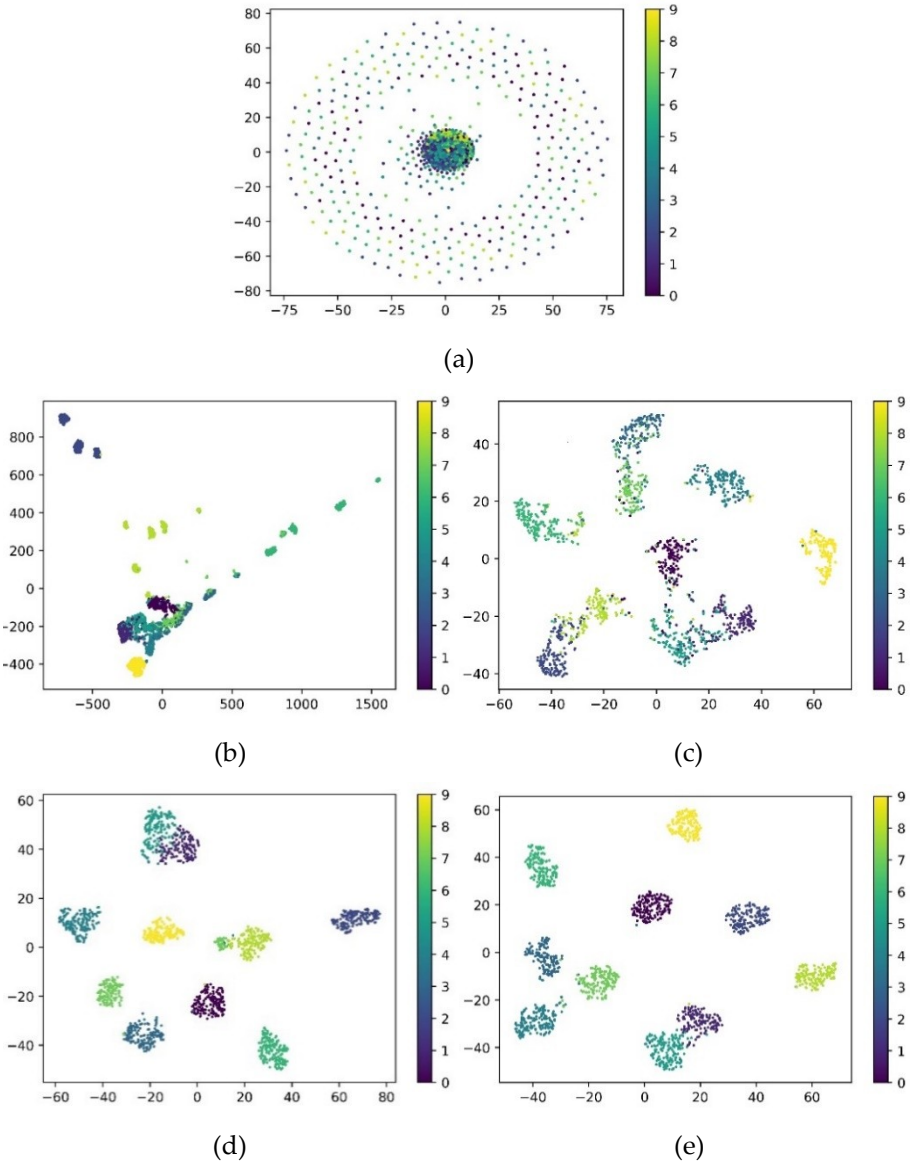


	-6	-4	-2	0	2	4	6	8	10	SNR/db
MSK-1DCNN	92.55	97.27	99.03	99.53	99.58	99.64	99.66	99.68	99.72	
WDCNN	86.15	91.85	95.15	97.19	98.23	98.76	98.93	99.2	99.22	
MLP	77.32	83.08	87.17	89.04	90.61	91.19	91.72	92.02	92.06	
SAE	70.34	78.8	84.06	86.59	87.86	88.86	89.37	89.46	89.57	

Figure 6. Comparison of diagnosis accuracy

To further demonstrate the performance of proposed MSK-1DCNN model, we use t-SNE (t-distributed stochastic neighbor embedding) algorithm to visualize the output of the last layer of each model shown above. The output Dimension of each model on testing set is reduced using t-SNE algorithm when the SNR is -4, and the results are displayed in a two-dimensional space, as shown in Figure 7. The fault state of original input signal is chaotic and inseparable, for the features are gathered into a distinguishable state after the feature extraction of each model. It can be seen that the output features of SAE are poorly aggregated. Although the various features of BP neural network are gathered together, the features overlap with each other and are not completely separated. The output of WDCNN and MSK-1DCNN have only a few fault states overlapping, and basically achieve separation of each fault state. It shows that these two models based on one-dimensional convolutional neural network have stronger feature extraction capabilities. Moreover, the feature aggregation degree of MSK-1DCNN is obviously better than WDCNN, which proves that the diagnostic performance of MSK-1DCNN model under strong noise is better than WDCNN.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 6 August 2020 doi:10.20944/preprints202008.0164.v1



(a) (b) (c) (d) (e)

Figure 7. Visualization of the input extracted from the last layer of different test models via t-SNE method: (a)raw input signals; (b)SAE; (c)BP Neural Network; (d)WDCNN; (e)MSK-1DCNN.

5. Conclusions

Aiming at the problem that current models have low fault diagnosis accuracy under noisy conditions, we propose a one-dimensional convolutional neural network named MSK-1DCNN. MSK-1DCNN uses a multi-scale convolution structure to extract different fault features from original signal and use ELU function instead of the ReLU function in MSK. At the same time, we use a training set with pepper noise to train MSK-1DCNN. MSK-1DCNN can extract high-efficiency discriminant features from original signal, and realize high-precision fault diagnosis of bearings under noisy conditions. According to the experiment, the conclusions of this paper are as follows:

（1）Through the multi-scale convolution structure, MSK-1DCNN can extract different discriminative features from original signal, so that it can obtain better diagnostic results at low SNR than the network using a single-layer and single-convolution kernel network.

（2）Using ELU activation function instead of ReLU function in MSK can effectively improve the anti-noise ability and increase the diagnostic accuracy of the at low SNR.

（3）Adding pepper noise to training data during training phase of the network can effectively destroy the input data, so that MSK-1DCNN can extract more essential features from original data and improve the generalization ability of network.

（4）The diagnostic accuracy of MSK-1DCNN at low SNR is significantly higher than other comparison models. The diagnostic accuracy at SNR is -6 is 92.55%, and the diagnostic accuracy at SNR is -4 is 97.27. Moreover, the diagnostic accuracy at higher SNR is also higher than other models. Compared with other models, MSK-1DCNN has better anti-noise performance.

This paper focuses on the problem of bearing fault diagnosis under noise conditions. The experimental data is the fault data generated under three fixed loads. In the actual working environment, the load of bearing often changes with the change of production task. The data obtained is variable working condition data under different loads. Therefore, in future works, we consider applying the research results of this paper to the fault diagnosis of bearings under variable conditions.

Funding: This research was funded by National Natural Science Foundation of China [Grant No.61763028]

References

1. Tang-Duy Hoang, Hee-Jun Kang. a survey on deep learning based bearing fault diagnosis. *Neurocomputing*, **2019**, *335*, 327-335. DOI: 10.1016/j.neucom.2018.0 6.078.
2. C Szegedy, S Ioffe, V Vanhoucke, et al. Inception-v4, Inception-Resnet and the Impact of Residual Connections on Learning. *arXiv preprint*, **2016**, arXiv: 1602.07261.
3. T Mikolov, M Karafiát, L Burget, et al. Recurrent neural network based language model. INTERSPEECH **2010**. DOI: 10.1109/EIDWT.2013.25.
4. H Greenspan, B van Ginneken, R M Summers. Guest editorial deep learning in medical imaging: overview and future promise of an exciting new technique. *IEEE Transactions on Medical Imaging*, **2016**, *35 (5)*, 1153–1159. DIO: 10.1109/TMI.2016.2 553401.
5. Xia Min, Li Teng, Liu Lizhi, et al. Intelligent fault diagnosis approach with unsupervised feature learning by stacked denoising autoencoder. *IET Science, Measurement & Technology*, **2017**, *11(6)*, 687-695. DOI: 10.1049/iet-smt.2019.0423.
6. Wei Zhang，Chuanhao Li，Gaoliang Peng, et al. A deep convolutional neural network with new training methods for bearing fault diagnosis under noisy environment and different working load. *Mechanical Systems and Signal Processing*, **2018**, *100*, 439-453. DOI: 10.1016/j.ymssp.2017.06.022.
7. H Shao, H Jiang, F Wang, et al. An enhancement deep feature fusion method for rotating machinery fault diagnosis. *Knowledge-Based Systems*, **2017**, *119*, 200-220. DOI: 10.1016/j.knosys.2016.12.012.
8. T Ince, S Kiranyaz, L Eren, et al. Real-time motor fault detection by 1-d convolutional neural networks. *IEEE Transactions on Industrial Electronics*, **2016**, *63 (11)*, 7067–7075. DOI: 10.1109/TIE.2016.2582729.
9. Levent Eren. Bearing Fault Detection by One-Dimensional Convolutional Neural Networks. *Mathematical Problems in Engineering*, **2017**. DOI: 10.1155/2017/8617315.
10. Wei Zhang, Gaoliang Peng, Chuanhao Li, et al. A New Deep Learning Model for Fault Diagnosis with Good Anti-Noise and Domain Adaptation Ability on Raw Vibration Signals. *Sensors*, **2017**, *17(3)*, 425. DOI: 10.3390/s17020425.
11. Jake Bouvrie. Notes on Convolutional Neural Networks. *MIT CBCL Tech Report*, Cambridge, MA, **2006**.
12. Nair V，Hinton G E，Farabet C．Rectified linear units improve restricted Boltzmann machines. *Proceedings of the 27th International Conference on Machine Learning*. **2010**, 807-814.
13. D-A Clevert, Thomas Unterthiner, Sepp Hochreiter. Fast and accurate deep network learning by exponential

linear units (elus). *arXiv preprint*, **2015**, arXiv:1511.07289.

14. Shuzhan Huang, Jian Tang, Juying Dai, el al. 1DCNN Fault Diagnosis Based on Cubic Spline Interpolation Pooling. *Shock and Vibration*, **2020**, *1949863*. DOI: 10.1155/20 20/1949863.
15. Wensi Tang, Guodong Long, Lu Liu, et al. Rethinking 1D-CNN for Time Series Classification: A Stronger Baseline. *arXiv preprint*, **2020**, arXiv:2002.10061v1.
16. Yong Yao, Sen Zhang, Suixian Yang, et al. Learning Attention Representation with a Multiscale CNN for Gear Fault Diagnosis under Different Working Conditions. *Sensors*, **2020**, *20(4)*, 1233. DOI: 10.3390/s20041233.
17. Boqing Zhu, Changjian Wang, Feng Liu, et al. Learning Environment Sounds with Multi-Scale Convolution Neural Network. *arXiv preprint*, **2018**, arXiv:1803.10 219v1.
18. Zhenyao Zhu, Jesse Engel, Awni Hannun. Learning Multiscale Feature Directly From Waveforms. *arXiv preprint*, **2016**, arXiv: 1603.09509v2.
19. Sergey Ioffe, Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv preprint*, **2015**, arXiv: 1502.03167v3.
20. Johan Bjorck, Carla Gomes, Bart Selman. Understanding batch normalization. *arXiv preprint*, **2018**, arXiv: 1806.02375v4.
21. Chuanhao Li, Wei Zhang, Gaoliang Peng, et al. Bearing Fault Diagnosis Using Fully-Connected Winner-Take-All Autoencoder. *IEEE Access*, **2018**, *6*, 6103-6115. DOI: 10.1109/ACCESS.2017.2717492.
22. Diederik Kingma, Jimmy Ba. Adam: A Method for Stochastic Optimization. *arXiv preprint*, **2015**, arXiv: 1412.6980v8.
23. Shao Haidong, Jiang Hongkai, Zhao Huiwei, et al. A novel deep autoencoder feature learning method for rotating machinery fault diagnosis. *Mechanical Systems and Signal Processing*, **2017**, *95*, 187-204.
24. Chen Lu, Zhen-Ya Wang, Wei-Li Qin, et al. Fault diagnosis of rotary machinery components using a stacked denoising autoencoder-based health state identification. *Signal Processing*, **2017**, *130*, 377-388. DOI: 10.1016/j.sigpro.2016.07.028.