Received 28 October 2021
Revised 22 January 2022
10 March 2022
Accepted 13 March 2022

DRN-GAN: an integrated deep learning-based health degradation assessment model for naval propulsion system

Jingtong Gao
School of Electric and Electronic Engineering, Nanyang Technological University, Singapore, Singapore and
School of Automation Science and Electrical Engineering, Beihang University, Beijing, China
Shaopeng Dong
School of Automation Science and Electrical Engineering, Beihang University, Beijing, China and
Ningbo Institute of Technology, Beihang University, Ningbo, China
Jin Cui
Research Institute for Frontier Science, Beihang University, Beijing, China and
Ningbo Institute of Technology, Beihang University, Ningbo, China, and
Mei Yuan and Juanru Zhao
School of Automation Science and Electrical Engineering, Beihang University, Beijing, China and
Ningbo Institute of Technology, Beihang University, Ningbo, China

Abstract

Purpose – The purpose of this paper is to propose a new deep learning-based model to carry out better maintenance for naval propulsion system.

Design/methodology/approach – This model is constructed by integrating different deep learning algorithms. The basic idea is to change the connection structure of the deep neural network by introducing a residual module, to limit the prediction output to a reasonable range. Then, connect the Deep Residual Network (DRN) with a Generative Adversarial Network (GAN), which helps achieve data expansion during the training process to improve the accuracy of the assessment model.

Findings – Study results show that the proposed model achieves a better prediction effect on the dataset. The average performance and accuracy of the proposed model outperform the traditional models and the basic deep learning models tested in the paper.

Originality/value – The proposed model proved to be better performed naval propulsion system maintenance than the traditional models and the basic deep learning models. Therefore, our model may provide better maintenance advice for the naval propulsion system and will lead to a more reliable environment for offshore operations.

Keywords Condition based maintenance, Deep learning, Deep neural networks, Naval propulsion

Paper type Research paper

1. Introduction

Maintenance of dynamic systems is quite an important work. Ensuring the safe operation of large mechanical systems can not only guarantee the economic interests of the system owners, but also greatly avoid the occurrence of unexpected safety accidents. However, the

This work was supported by the Beijing Natural Science Foundation under Grant L212033.

Engineering Computations
Vol. 39 No. 6, 2022
pp. 2306-2325

0264-4401
DOI 10.1108/EC-10-2021-0624

maintenance of complex systems often requires a high level of expertise, which is difficult (Kobbacy *et al.*, 2008) and costly to achieve (Castro *et al.*, 2020).

At present, the main equipment maintenance strategies include Corrective Maintenance (CM) (Özgür-Ünlüakın *et al.*, 2021) and Preventive Maintenance (PM). The CM methods can only repair or replace equipment when it fails to operate, so the implementation of these methods cannot avoid the cost and revenue loss associated with system anomalies (Kothamasu and Huang, 2007). PM methods are used to predict the exact moment when equipment needs replacement or maintenance (Chang, 2014). One of the main methods in PM, called PRedetermined Maintenance (PRM), uses the average life of a type of equipment in the system to predict the current working state of the same equipment (Selvik and Aven, 2011). It can reduce the occurrence of equipment failure in a certain sense, but it cannot guarantee effective prediction of fault behavior. Another PM method named Condition-Based Maintenance (CBM) (Ahmad and Kamaruddin, 2012) methods are a group of methods mainly in use nowadays, which aim to conduct health degradation assessment by directly predicting equipment performance decay coefficients using information about the current system. The CBM methods' effectiveness is positively correlated with the prediction accuracy of the related algorithms. Now, with the progress of computing technology and the proposal of different deep learning methods, it is possible to build a high-precision CBM model based on deep learning algorithms to effectively predict the equipment performance decay coefficients, so as to improve the efficiency and reliability of the system.

In this paper, we analyze the different representative CBM methods for health degradation assessment of a COmbined Diesel-eLectric And Gas (CODLAG) propulsion system used in naval vessels, which fall into two categories: traditional CBM methods, like Hidden Markov Model (HMM), and CBM methods based on deep learning algorithms, like RandomForest etc. According to the test results and some other deep learning algorithms, a new model named DRN-GAN is proposed through adjustment and improvement of Deep Neural Network (DNN) framework. This model is mainly used in the naval propulsion system maintenance scenario represented by the dataset used in this paper. The new model solves the problem of the difficult deep information transmission process in DNN by using residual module that is originally used in Convolutional Neural Networks (CNN). By doing this, the overall efficiency and accuracy of the model are improved. And, by restricting the output value to a certain range, the new model ensures the practical significance of the prediction outputs and in the meantime its accuracy has been improved. In addition, the model is put into a generative adversarial network to help further improve its accuracy on small datasets by providing data expansion during the training period. The results show that DRN-GAN is more accurate in terms of overall performance than other CBM methods, and more effective for the CODLAG propulsion system.

The rest of the thesis is organized as follows. In Section 2, the dataset of the CODLAG propulsion system and different CBM methods are introduced. In Section 3, effective data processing methods are carried out on the dataset to improve the training efficiency. Then, in Section 4, several traditional CBM methods and CBM methods based on deep learning are modeled and tested. In Section 5, based on some problems found in the tests and a reference to their solutions, the DRN-GAN model is constructed and tested. Finally, in Section 6, the performance and accuracy of the above models on the dataset are compared to verify the conclusion that the DRN-GAN model has a better prediction effect on the dataset. Concluding remarks and issues requiring further study are summarized in Section 7.

2. Naval propulsion system dataset and maintenance methods

In this paper, CBM methods for naval propulsion system are studied. In order to better evaluate different CBM methods and design a better maintenance model, it is necessary for us to understand CODLAG propulsion system and the existing different CBM methods.

2.1 Naval propulsion system and dataset

2.1.1 Naval propulsion system. Naval forces in many countries use the CODLAG propulsion system on their frigates or destroyers. The European Multi-Mission FRigate (FREMM) used by the Italian Navy, the F125 Baden-Wurttemberg class frigate used by the German Navy, and the Type 23 frigate used by the United Kingdom Royal Navy all implement the CODLAG propulsion system (Pal *et al.*, 2019). A brief diagram of CODLAG propulsion system is shown in Figure 1. It is a hybrid propulsion system (Martelli and Figari, 2017) using Gas Turbines (GTs) when high speed is needed. An electric motor driven by a diesel generator is connected to a transmission shaft. GTs use cross-connected gearboxes to rotate the shaft when high speeds are needed. Systems such as the propulsion units of naval destroyers should always be in working order without interruption, and their performance should be continuously monitored to avoid any unexpected failures (Doerry *et al.*, 2015; Borisov *et al.*, 2006). In this case, both CM methods and PRM methods cannot avoid the interruption, loss, and safety problems caused by failures. On the contrary, CBM methods are good maintenance strategies since they can predict equipment performance decay coefficients continuously to help avoid system failures. And sensors integrated into system components can provide real-time streaming data for risk prediction of potential failures without interrupting system work and performance (Hanson, 2011).

2.1.2 Naval propulsion system dataset. This paper collects data from University of California, Irvine (UCI) machine learning knowledge base (Asuncion and Newman, 2007). The data are generated by a complex GT simulator mounted on the frigate whose propulsion system is a CODLAG propulsion unit (Coraddu *et al.*, 2016). The different modules that form the full simulator have been developed and fine-tuned on several similar real-world propulsion devices over the years. In view of these observations, the available data are consistent with the possible true ship. In this paper, CBM methods are adopted to predict two equipment performance decay coefficients of GT by analyzing 16 indicators representing the system state (Cipollini *et al.*, 2018a).

The input vectors with 16 characteristics contain the parameters shown in Table 1, which affect the performance of the GT and its compressor and may lead to the decay of their coefficients.

The modeling goal is to extend the life expectancy of equipment by accurately predicting the two output coefficients shown in Table 2: Gas Turbine Compressor decay (GTC decay) and Gas Turbine decay (GT decay). They are uniformly distributed in the dataset (Pal *et al.*, 2019).

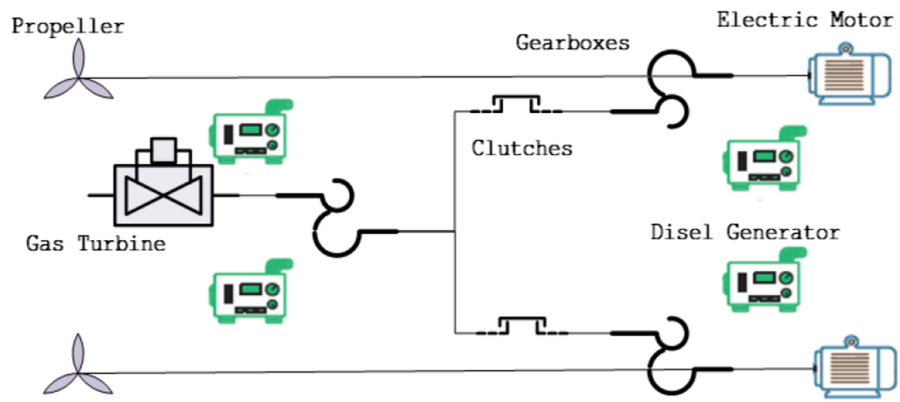


Figure 1.
CODLAG propulsion system

#	Variable name	Unit
1	Lever position (lp)	[ ]
2	Ship speed (v)	[knots]
3	Gas Turbine shaft torque (GTT)	[kN m]
4	Gas Turbine rate of revolutions (GTn)	[rpm]
5	Gas Generator rate of revolutions (GGn)	[rpm]
6	Starboard Propeller Torque (Ts)	[kN]
7	Port Propeller Torque (Tp)	[kN]
8	HP Turbine exit temperature (T48)	[C]
9	GT Compressor inlet air temperature (T1)	[C]
10	GT Compressor outlet air temperature (T2)	[C]
11	HP Turbine exit pressure (P48)	[bar]
12	GT Compressor inlet air pressure (P1)	[bar]
13	GT Compressor outlet air pressure (P2)	[bar]
14	Gas Turbine exhaust gas pressure (Pexh)	[bar]
15	Turbine Injection Control (TIC)	[%]
16	Fuel flow (mf)	[kg/s]

Table 1. Input data

#	Variable name	Unit
1	GT Compressor decay state coefficient	[ ]
2	GT Turbine decay state coefficient	[ ]

Table 2. Output data

2.2 Relevant CBM methods

CBM methods are generally based on a large amount of actual or simulated data. So, the models of CBM methods are also called data-driven models. They are developed to predict equipment performance decay coefficients (Berry and Linoff, 2004) by finding relationships between output values and input values. The operators would set a threshold previously and could thus determine the time of maintenance and replacement of equipment according to its performance decay coefficients. Therefore, when other conditions remain unchanged, the higher the accuracy of the CBM models, the better the maintenance effect of CBM methods will be. The CBM methods are described in Figure 2.

At present, there are many kinds of CBM algorithms. Many CBM algorithms suitable for different systems can be found in the study of Jardine *et al.* (2006), among which the most commonly used is the Support Vector Machine (SVM) (Widodo and Yang, 2007), HMM (Carey *et al.*, 2000) and Kalman filter (Simani *et al.*, 2003). In addition, the kernel-based method is applied in the study of Coraddu *et al.* (2016). Image processing technology is applied in the research done by Bagavathiappan *et al.* (2013). In the study of Basurko and Uriondo (2015), an artificial neural network is used to predict the state of the engine and propeller, and a complete overview of the relevant technology can be found in the study of Lazakis *et al.* (2016), and this work can be seen as a continuation of the study of Coraddu *et al.* (2016). Moreover, CBM method research works on a dataset similar to the one mentioned in this paper are done by Tan *et al.* (2019) and Cipollini *et al.* (2018b).

In the dataset of naval propulsion system described in this paper, the input dimension of the dataset is small, the data volume of this dataset is small, and the output are continuous values. Therefore, the appropriate traditional CBM algorithms are Support Vector Regression (SVR) (Smola and Schölkopf, 2004) and HMM, and the appropriate CBM algorithms based on deep learning are Adaboost (Freund and Schapire, 1997), RandomForest (Breiman, 2001) and Deep Neural Networks (DNN) (Manikandan and Duraivelu, 2021).

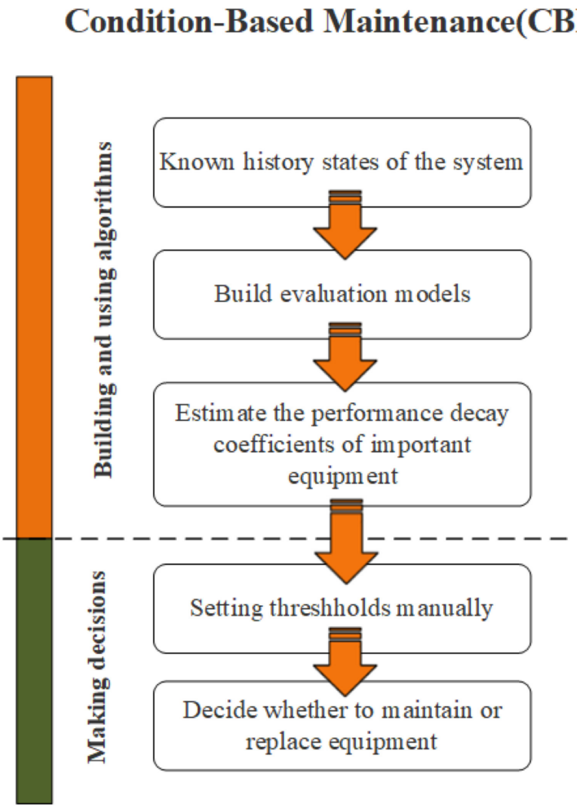


Figure 2.
Condition-based maintenance

These algorithms are overall modeled by relevant mathematical principles with the exception of DNN, so it is difficult to adjust the models for different systems without retraining their mathematical principles. Therefore, this paper would only adjust the superparameters when modeling these methods to test their optimal accuracy and performance.

On the other hand, the mathematical principle of DNN algorithm only provides the neural framework for constructing a network, so it is feasible to adjust and improve DNN networks for specific problems. In this paper, we will first choose the appropriate DNN frameworks and then try to build a more accurate model by adjusting and improving DNN frameworks.

3. Data processing

Data processing can shorten the training time and improve the training effect by screening and transforming the data in the dataset (Burdack *et al.*, 2020). And it can help improve the modeling efficiency and the performance and precision of the final model to a certain extent.

3.1 Data processing process

In this paper, by observing the expression form of data in the dataset, the feature clipping behavior is carried out, which can reduce the amount of data without reducing the effective information. At the same time, normalization method, Principal Component Analysis (PCA) and random noise enhancement algorithms may be used as methods of data preprocessing.

The effect of the above algorithms are tested on the dataset, and algorithms with positive effects are used for data processing. After data processing, the dataset is divided into train set and test set in a ratio of 7:3. The data processing process is shown in Figure 3.

3.2 Model evaluation criteria

This paper mainly evaluates the performance and accuracy of models on the dataset from two aspects, namely Mean Squared Error Loss (MSE Loss) and Determination Coefficient (R^2). Here, GTCloss and GTloss represent the MSE loss of the two outputs; Aveloss represents the arithmetic mean of above two MSE losses. R2-GTC and R2-GT represent the R^2 coefficients of the two outputs, and R2-Ave represents the arithmetic mean of above two R^2 coefficients. The two evaluation indicators are defined as follows:

(1) MSE Loss

The MSE Loss indicator is defined in Equation (1), where $y(x)$ is the actual output, $\hat{y}(x)$ is the predicted output, and n is the total number of samples.

$$MSELoss = \frac{1}{2n}\sum_{x}\|y(x) - \hat{y}(x)\|^2 \tag{1}$$

It represents the difference between the prediction and the actual output. The smaller the MSE Loss, the more accurate the fitting result should be.

(2) Determination Coefficient R^2

R^2 coefficient is defined in Equation (2), where y_i is the actual output value, y_{pre} is the predicted value, and $\bar{y}$ is the mean value of the actual output.

$$R^2 = 1 - \frac{\sum(y_i - y_{pre})^2}{\sum\left(y_i - \bar{y}\right)^2} \tag{2}$$

It represents the fitting effect of data and the utilization degree of valid content of data. Its value range is $(-\infty, 1]$. The larger the value is, the better the fitting effect is and the higher the utilization degree of valid content of data is.

3.3 Feature clipping

In the selected dataset, column 9 represents GT Compressor Inlet air temperature, and column 12 represents GT Compressor Inlet air pressure. The two features are constant on the whole dataset, and their images are displayed as a horizontal line segment. So they are judged as invalid features. The purpose of feature clipping is to remove these features. After feature clipping, the input feature dimensions change from 16 to 14. The characteristic image of column 9 is shown in Figure 4a, and the characteristic image of column 12 is shown in

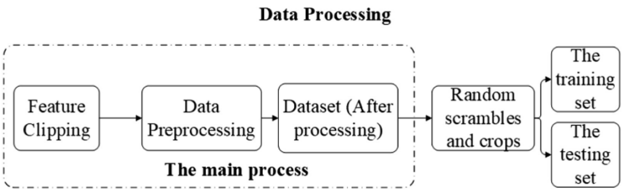
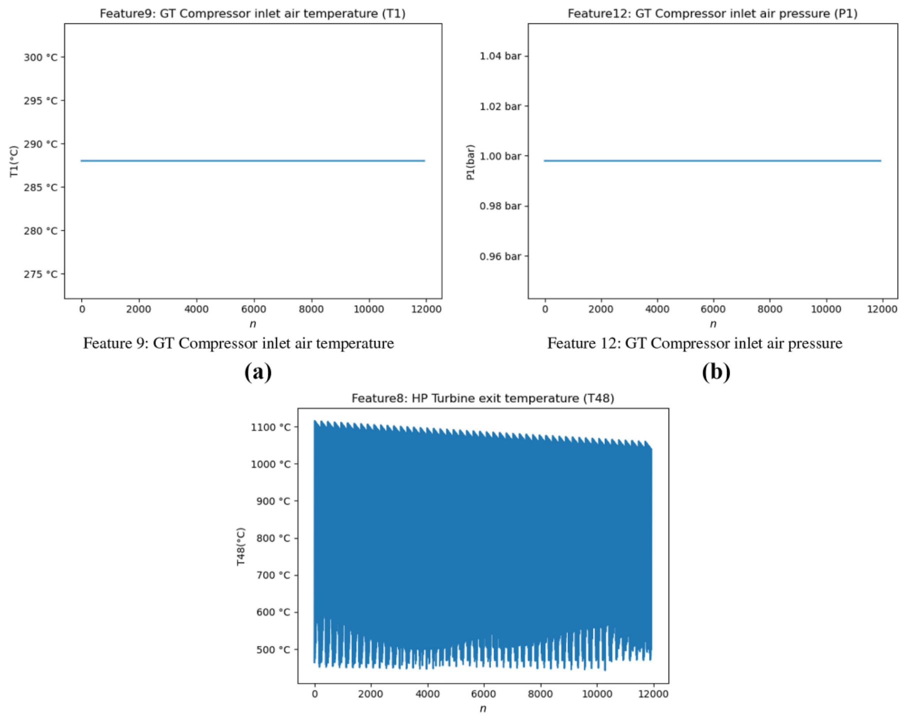


Figure 3. Data processing



Feature 9: GT Compressor inlet air temperature
(a)

Feature 12: GT Compressor inlet air pressure
(b)

Feature 8: HP Turbine exit temperature
(c)

Figure 4.
Feature 9, 12 and 8 of UCI CBM dataset

Figure 4b. As a comparison, the characteristic of column 8 representing high-pressure Turbine exit temperature (HP Turbine exit temperature) is shown in Figure 4c. The value of this feature fluctuates on the whole dataset.

3.4 Data preprocessing

After feature clipping, the data still have uneven distribution, and repeated training is easy to overfit. Therefore, we test normalization, PCA, noise enhancement and other preprocessing methods, and use methods beneficial to model training so as to provide a basis for efficient model training. Here, the SVR model was selected as the test model, and the average result of ten tests results is taken as the final result. The processing results of these methods are shown in Table 3.

Table 3.
Data preprocessing effect (SVR model)

Data preprocessing	GTCLoss	GTLoss	R2-GTC	R2-GT
(>1 dim) PCA	>7e-6	>7e-6	<0.985833	<0.940217
(1 dim) PCA	7.00E-06	7.00E-06	0.985833	0.940217
Raw result	4.00E-06	6.00E-06	0.990936	0.941512
Norm	9.00E-07	5.00E-06	0.994719	0.943074
Norm & 1e-5 Noise	5.00E-07	5.00E-06	0.997156	0.943018
Norm & 1e-6 Noise	9.00E-08	4.00E-06	0.999503	0.945888
Norm & 1e-7 Noise	4.00E-07	4.00E-06	0.996586	0.944917

Normalization method converts input feature data into data with mean value of 0 and variance of 1. This method changes different feature distributions into standard normal distribution, which is conducive to gradient descent of the model during training and the improvement of the final model accuracy. The normalization formula is shown in Equation (3). It can be seen from the processing results that the accuracy of the model is improved after normalization.

$$x_1 = \frac{(x - \mu)}{\sigma} \tag{3}$$

PCA method uses the size of feature as the basis of feature importance and ranks the feature importance. By dropping the following several features, it achieves the purpose of saving important features as far as possible and reducing the calculation amount. The mathematical expression of this process is shown in Equation (4). Where, m is the sample size, n is the sample feature dimension, k is the target feature dimension, *DataAverage* is the data subtracted from the mean value, *FeatureVectors* is the matrix composed of arranged Feature Vectors, and *ProcessedData* is the data after the final projection. Since there are only 14 valid features in the current dataset, PCA method will significantly reduce the model prediction effect.

$$\begin{aligned} &ProcessedData(m \times k) \\ &= DataAverage(m \times n) \times FeatureVectors(n \times k) \end{aligned} \tag{4}$$

In the random noise enhancement method, the random noise with small changes is added to make the sample value of the dataset fluctuate within a certain range during each cycle traversal. In this way, the stability of the network for the prediction results can be strengthen and the accuracy of model can be improved. The numerical range of random noise is a superparameter in network training. It can be seen from the test results that, the random noise with the size about 1e-6 is used for the normalized data to achieve the best effect.

In summary, the test results show that the normalization method and 1e-6 noise enhancement are most conducive to the training of the model, while PCA algorithm will lose a large amount of data prediction accuracy. So, the normalization and 1e-6 noise enhancement are used to enhance the data. The model verification shows that the processed data can be trained more quickly and efficiently.

4. Modeling and testing of different CBM methods

In this section, we discuss how to model different CBM methods based on the dataset in this article and test their effectiveness. Here, all the algorithms are written by Python and are rewritten by introducing basic functions in PyTorch, HMMLearn and SkLearn packages to meet the structural modifications in this paper. In addition, Equipment used for the experiments is a PC with Intel Core i7-7700HQ CPU 2.80 GHz with 8 GB RAM and NVIDIA GTX 1050Ti.

4.1 Modeling of different CBM methods

4.1.1 HMM. In the HMM model, the system state, as an important parameter of the system, is not directly visible, but the output depends on the state and the input is visible. Thus, the system state can be obtained by observing the output of the system. Through the distribution of output, the state transition mode of the system can be further estimated, and the output of the system under other conditions can be predicted.

HMM is generally used for classification tasks (Carey *et al.*, 2000). Since the range of output of the dataset in this paper is continuous, so we further change the classification model to a fitting model: In this paper, the output is classified according to the value size. The classification process is shown in Figure 5. The number of classes exists as a superparameter. Each data is divided into the class where the nearest point is, and then classification prediction is made. The final output of each class is the value of such central point.

4.1.2 SVR. SVM was originally used for binary classification. With the continuous development of this algorithm, SVM model has also been extended to a variety of problems, and SVR (Smola and Schölkopf, 2004) is one of the main algorithms of SVM used to solve the fitting problems. From a mathematical point of view, the goal of SVR algorithm is to find the plane expression that minimizes the total difference between all sample points and the high-dimensional plane constructed by data features.

In order to fit the nonlinear function, SVR model adopts the kernel function (also known as the kernel technique) (Coraddu *et al.*, 2016). This process can also be regarded as the process of mapping low-dimensional nonlinear features into higher-dimensional linear features. In application, the Radial Basis Function (RBF) (Wang *et al.*, 2013) is the dominant kernel Function, and the expression of RBF is shown in Equation (5). Where, x, x_i are the coordinates of two points in the space, δ is the bandwidth, which is used to control the longitudinal effective interval of the function.

$$k(x, x_i) = \exp\left(-\frac{\|x - x_i\|^2}{2\delta^2}\right) \tag{5}$$

4.1.3 RandomForest. RandomForest model uses Bagging method (Breiman, 1996) to build its sub-dataset for each learner to carry out training and learning process, and the final output is the predicted average of all learners. The learner uses the decision tree model. Here, each leaf node represents a classification, and each classification divides the dataset into two parts according to the current value of the most obvious feature. Finally, the output mean of the original data in the leaf node into which the input data are classified is calculated, which is the predicted output of the input data of this item.

4.1.4 Adaboost. Adaboost adjusts the dataset by updating the weight of the samples. After training a learner, more weights will be added to samples where the predicted value deviates too far from the actual value. The final prediction result is determined by the weighted combination of the output results of all learners (Koduri *et al.*, 2019). The learners also use the decision tree model as the main algorithm.

4.1.5 DNN. The DNN model is mainly composed of input, output and multiple hidden layers. Each layer is composed of several "neurons", among which, each neuron represents a nonlinear function composed of linear function and nonlinear activation function. The whole network adjusts the weight of these nonlinear functions through the difference between the model outputs and the real results, and the error is propagated back so that the functions can fit the complex nonlinear functions through different weighted combinations. A DNN model with several hidden layers is represented by Figure 6. The mathematical expression of the neuron is shown in Equation (6), where x_{ij} is the neuron j in layer i, w_{ij} is its weight, b_{ij} is its bias and $\sigma()$ is the nonlinear activation function.

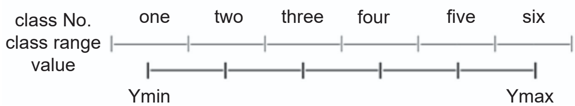


Figure 5.
Output classification

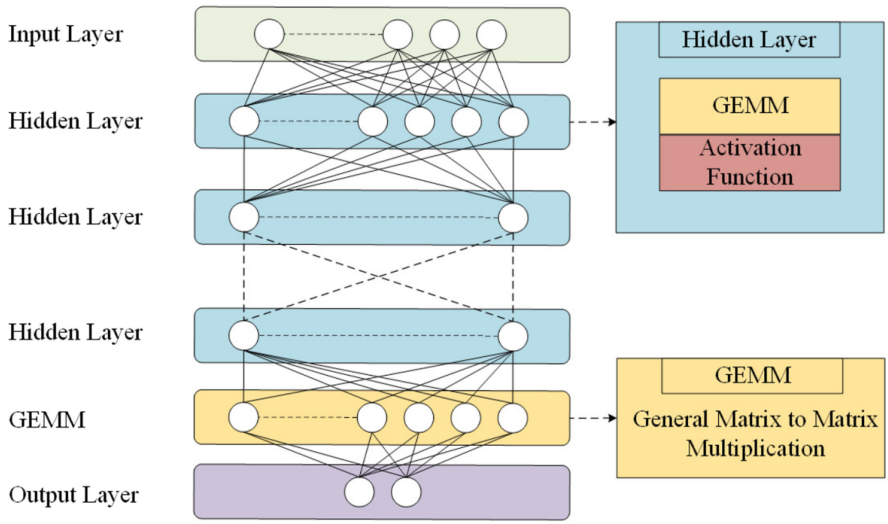


Figure 6.
DNN model

$$\begin{cases} \tilde{x}_{i+1,j} & = \sum_i (\omega_{ij} x_i + b_{ij}) \\ x_{i+1,j} & = \sigma\left(\tilde{x}_{i+1,j}\right) \qquad ((i+1) \ \textit{is not output layer}) \\ x_{i+1,j} & = x_{i+1,j} \qquad ((i+1) \ \textit{is output layer}) \end{cases} \tag{6}$$

Commonly used activation functions include ReLU (Płaczek and Płaczek, 2018; Glorot *et al.*, 2011), Sigmoid, Tanh, etc. (Dhumale, 2018). After parameter adjustment experiments in the early stage, ReLU function has a better effect under the dataset in this paper, thus the ReLU function is used as the activation function in this paper. In this paper, the number of neurons in DNN networks and the number of network layers are adjusted as superparameters.

4.2 Test results

We model the above methods. After several parameter adjustments for each model, a set of parameters that makes the model with the highest accuracy among several tests is selected. Then, the average value of the test results of the 10 groups is taken as the final evaluation. In this process, we select three representative structures for DNN: low-neuron model (DNN1: 8-6-4-2), model with a few layers (DNN2: 64-128-16-2) and model with many layers (DNN3: 16-32-32-32-16-2) for testing to compare the influence of different structures on the prediction accuracy. It is worth noting that, The low-neuron DNN model was derived from the study of Pal *et al.* (2019) on the same dataset, and the MSE loss of this model is basically consistent with the test results of Palash Pal *et al.*. The first seven rows of Table 4 shows the values of evaluation coefficients of each model.

The results show that the SVR algorithm is very effective and accurate in predicting GTC decay among conventional CBM methods, with GTCLoss of 6.40e-08 and R2-GTC of 2.74e-06. However, its prediction of GT decay is not good enough. RandomForest and DNN3 also achieve good results on the dataset. Among the three DNN models, the MSE loss test results of the DNN1 are basically consistent with the research results of Pal *et al.* (2019). However,

Table 4.
Test results of different CBM models

Model	GTCLoss	GTLoss	AveLoss	R2-GTC	R2-GT	R2-Ave
HMM	8.78E-05	1.65E-05	5.22E-05	0.588892	0.702861	0.645877
SVR	*6.40E-08*	2.74E-06	1.40E-06	*0.999705*	0.950888	0.975297
Adaboost	1.49E-04	4.65E-05	9.78E-05	0.310422	0.181170	0.245796
RandomForest	9.75E-07	6.14E-07	7.95E-07	0.995419	0.988941	0.992180
DNN1	2.21E-04	5.55E-05	1.38E-04	−0.000175	−0.003696	−0.001936
DNN2	2.13E-06	1.79E-06	1.96E-06	0.990084	0.967904	0.978994
DNN3	4.43E-07	1.01E-06	2.72E-06	0.998479	0.975671	0.987075
DRN-Origin	2.17E-07	1.80E-07	1.99E-07	0.999006	0.996782	0.997894
DRN-Sig	3.62E-07	3.03E-07	3.33E-07	0.998273	0.994526	0.996400
DRN-Bound	1.44E-07	1.98E-07	1.71E-07	0.999335	0.996503	0.997919
DRN-GAN	9.50E-08	*4.10E-08*	*6.80E-08*	0.999567	*0.999263*	*0.999415*

according to the R^2 coefficients, the actual prediction effect of this model is not good enough. At the same time, it can be observed that with the deepening of the models and the increase of the number of neurons, the prediction effect of DNN models is better and the accuracy is higher. In addition, it is worth noting that the R^2 coefficients of DNN1 are negative and close to zero. An obvious case when R^2 coefficients are zeros is that the output takes the mean of all outputs. So, it can be said that DNN1 achieves a mean prediction effect, or even worse.

5. Improved model construction based on DNN

In the models described in Section 4, with the exception of DNN, other algorithms are built by the overall modeling of relevant mathematical principles. Therefore, only parameter adjustments are made in the modeling of these methods in this paper to test their accuracy and performance. However, mathematical principle of algorithm within DNN provides only a framework. Therefore, building a network for adjustment and improvement in view of the specific issues is feasible. Through the test and analysis in Section 4, it can be seen that DNN with deeper layers and more neurons has a better fitting effect. Therefore, this section further uses some methods to construct a more accurate model by adjusting and improving DNN models with deeper layers and more neurons.

5.1 Structural improvement

It can be seen from the model test results that DNN networks with more neurons have a better fitting effect on the dataset. However, with the deepening of the network layer, the DNN network will have the phenomenon of "gradient attenuation", which will slow down the training process of the network. At the same time, the attenuation of interlayer gradient will also slow the parameter updating and hinder the improvement of accuracy.

This phenomenon also exists in other deep learning models. At present, in the field of image recognition, a similar problem is solved by He *et al.* (2016), and the solution is called residual module or direct connection and the corresponding network is called DRN. In the paper, the author found that with the increase in the number of network layers of CNN (Krizhevsky *et al.*, 2012), the loss of train set would gradually decrease. After the loss reaches a critical value, when the network deepens, the loss of the train set gradually increases.

If the low-level feature can be transmitted to the higher level, the higher level can obtain more effective feature information. In the forward operation, as the number of layers deepens, the effective information contained in the feature graph will decrease. The addition of residual modules ensures that effective information can be spread more easily. At the same time, since the parameters of the residual module can be adjusted automatically in the training process, the structure does not deactivate the middle layer. Schematic diagram of the residual module is shown in Figure 7.

In this paper, the structure described by this method is applied to the DNN model. In the CNN network, the output of each layer of network can be represented as a 2D array, and the calculation process of the residual module is represented as the addition of the 2D arrays output by two different layers. In DNN network, each layer of network can be represented as a 1D array, and the calculation process of residual module can be represented as the addition of one-dimensional arrays output by two different layers.

By using residual module and training to adjust the superparameters for many times, this paper constructs a new model. For the convenience of description, we called it DRN-Origin. Its structure is shown in Figure 8. Among them, the output 2/1 means that we will construct a model predicting both outputs and two models predicting one of the two outputs separately. Then we will select models with the better test results for evaluation. This is because after several tests, we find out that the effects of models predicting two outputs are similar to the effects of models predicting one output on this dataset.

We train and test the DRN-Origin model. The results are shown in row 8 of Table 4. It can be seen from the results that the accuracy of the model has been significantly improved. The MSE Loss coefficients of both outputs reach about 1e-07, and the R^2 coefficients also reach

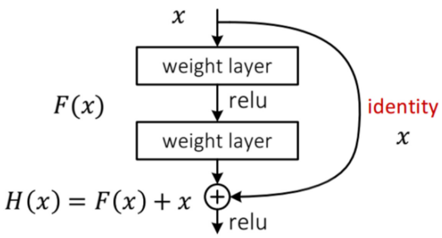


Figure 7.
Residual module

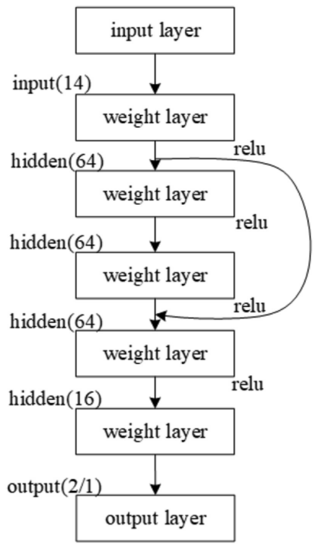


Figure 8.
DRN-Origin model

above 0.99. In the training process, it is noted that the training time is shortened by nearly half, and the training speed is also improved compared with the original DNN models.

5.2 Output improvement

The fitting output images of DRN-Origin model on the whole dataset are shown in Figure 9a and b. From the images, we can find out that the output of the model exceeds the upper and lower bounds. This phenomenon not only reduces the precision of the model but also does not correspond to the actual meaning of the data represented in the dataset. This is because the GTC decay and GT decay have definite boundaries and the corresponding physical meanings. The upper bound “1” represents the completely intact state of health. Obviously, it is not reasonable to predict a maximum value more than “1”. Since the dataset in this paper only needs to use the nondegraded and slightly degraded data to judge the equipment performance decay coefficients, the lower bound also has a given minimum value. Thus, the predicted data beyond the lower bound should also be considered unreasonable. These problems occur because the output range is not set in the current model. And there are two solutions to this situation.

5.2.1 Add sigmoid function. One effective solution is to add Sigmoid function to the network output layer, after which we can limit the output to the specified range by scaling and offset.

The mathematical expression of the Sigmoid function is shown in Equation (7). It can map data between (0, 1) to ensure that the data have upper and lower bounds.

$$y_{sigmoid} = \frac{1}{1 + e^{-x}} \tag{7}$$

After the Sigmoid function is added, the output is scaled and offset to make the function fit the data in the given interval. The output interval for GTC decay in this dataset is [0.95, 1] and for GT decay is [0.975, 1]. The processed model using this method is called DRN-Sig and the test results are shown in row 9 of Table 4. The fitting images of the model on the whole dataset are shown in Figure 10a and b.

From the fitting results, it can be seen that there has been a marked decrease in the number of models that exceed the threshold. However, by observing the output data, we find out that the output of the model is difficult to reach the value near the boundary at this time, and the predicted data will have certain bending at both the upper and lower bounds. This is because the output of the Sigmoid function itself only approaches the upper/lower bounds at the input

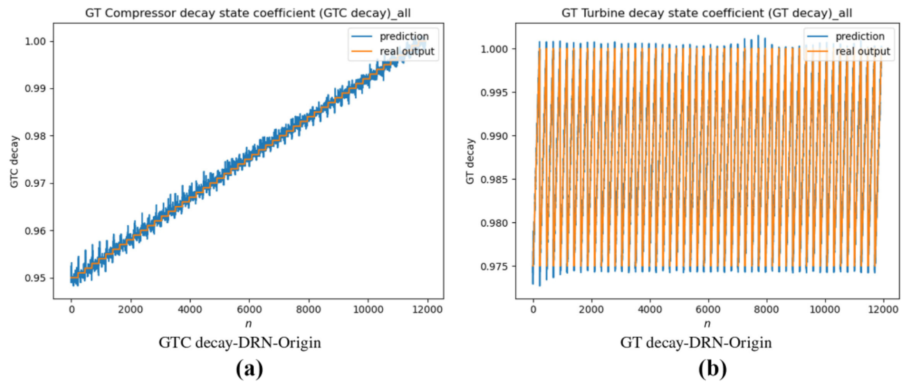


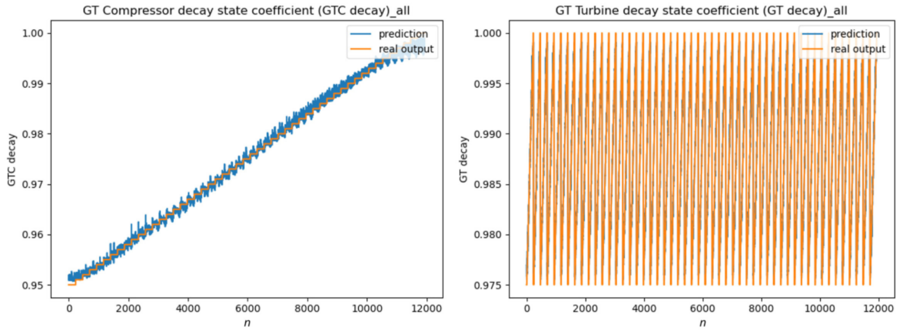
Figure 9.
DRN-Origin fitting outputs

infinity. Therefore, although the model avoids the situation of data transgressions, the accuracy of the model is not improved.

5.2.2 Add upper and lower bounds. The second approach is not to change the network structure but simply to add upper and lower bounds to the model's output layer.

The constraint of the output can be implemented not by the model itself but by adding upper and lower bounds to the predicted output data of the model at test time. This method not only ensures that the predicted output of the model will not go out of bounds under this dataset but also makes the model not be restricted by the range interval in the training process on other datasets. The modified model is called DRN-Bound in this article, and its test results are shown in row 10 of Table 4. The fitting images on the whole dataset are shown in Figure 11a and b.

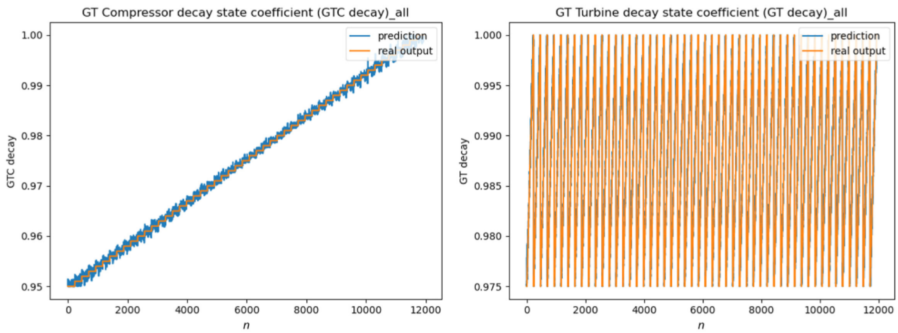
The results show that DRN-Bound improves its prediction accuracy while normalizing the output within the interval. It reduces the degree to which the predicted value deviates from the actual output by operating only on values that are outside the interval. At the same time, since the method of limiting interval in DRN-Bound is not related to model training and model testing process, further changes in model structure will not influence the effect of this method.



GTC decay-DRN-Sig
(a)

GT decay-DRN-Sig
(b)

Figure 10.
DRN-Sig fitting outputs



GTC decay-DRN-Bound
(a)

GT decay-DRN-Bound
(b)

Figure 11.
DRN-Bound fitting outputs

5.3 Improvement of training methods

The third problem comes from the dataset itself. In general, DNN networks work for large datasets, which typically have more than 100,000 sets of data, and the more data in the dataset, the better the fitting performance of the same DNN network. However, in general, equipment performance decay data require experienced experts to evaluate the decay of system components, so it is difficult to collect a large amount of data. Thus, how to enhance or expand the data content on the basis of limited data has become an important aspect of optimizing the accuracy of the model.

In the field of deep learning, data enhancement relies on data clipping, noise addition, sampling and other behaviors, which has been realized in the data preprocessing part. Another effective way to improve model accuracy is to increase the amount of data. However, due to the high cost and cumbersome process of data collection for equipment performance decay evaluation, it is obviously difficult for researchers to collect data spontaneously. In deep learning algorithms, in addition to artificially collecting data and increasing the amount of data from the dataset itself, another method to increase the amount of data is called data expansion.

GAN is generally used for data expansion (Salimans *et al.*, 2016), which is originally used to generate pseudo-real data. GAN network can be divided into two parts: generation model and discrimination model. The generation model generates fake samples, and the discrimination model tries to distinguish the fake samples from the generation model and the true samples from the train set. The generation model can learn the sample feature distribution, and the discrimination model can learn the authenticity of the data. Both generation model and discriminant model can use deep learning algorithms. Finally, their mutual game improves the accuracy of both. The generation model can generate pseudo-data that closely resembles the real one, and the discrimination model can also determine the authenticity of data with high accuracy. The schematic diagram of GAN network is shown in Figure 12.

Since GAN network can generate "pseudo-true" data, it can be used for data expansion. In the aspect of image recognition, data enhancement based on GAN network can be seen in the study of Odena *et al.* (2017), and the related network is called Auxiliary Classifier Generative Adversarial Network (ACGAN). It is important to note that this approach is generally effective on small datasets and not so effective on large datasets. Our dataset contains 11,934 sets of data, which is a small dataset. Therefore, adjusting the training process of the model with the antagonistic principle of GAN network is expected to further improve the model's accuracy.

Since the dataset in this paper corresponds to the fitting problem and the original ACGAN is applied to the classification problem, the loss function form needs to be changed to make this method adapt to our dataset. The specific method is as follows: Adding a 1D output feature through the Sigmoid function to judge the authenticity of the data on the basis of the original model. Then, the original model can be called the "discrimination model". Then we add a DNN model with random noise as input and original input characteristic dimension as

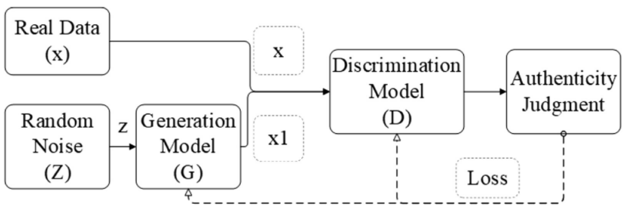


Figure 12.
GAN model

output, which is called the generation model. The generation model takes fitting data features as the goal, while the discrimination model takes fitting data output and discriminating data authenticity as the goal for training process.

The loss function of the discrimination model is shown in Equation (8).

$$Loss = MSELoss(dataset) + BCELoss(dataset, 1) + BCELoss(generator, 0) \tag{8}$$

Where, *BCELoss*(*generator*, 0) represents the dichotomy output of fake data through the discrimination network and calculates BCE loss with the label “0” (where “0” represents the “fake” label and “1” represents the “true” label) for identification of fake data. Similarly, *BCELoss*(*dataset*, 1) is used to identify true data. Meanwhile, the data generated by the generation model will also be output through the discrimination model, and the loss function of the generation model is calculated as shown in Equation (9).

$$Loss_1 = (generator, 1) \tag{9}$$

Finally, as the generation model approximates the original data features, the discrimination model will be able to better recognize the original data features. In this process, the train set of the discrimination model is expanded by the data generated by the generation model. Therefore, the accuracy of the final discrimination model is expected to be further improved. In this dataset, after testing, the DNN model of 64-64-128-64-14 is adopted for the generation model to achieve a better effect, and the DRN-Bound model is adopted for the discrimination model. The whole model is called DRN-GAN. The structure of DRN-GAN is shown in Figure 13.

The extended data amount generated by the generation model was used as a superparameter to adjust. When the extended data accounted for about one-fourth of the total dataset, the optimal optimization effect for the discrimination model could be achieved. It is important to note that when the expansion of data is more than three-quarters according to the set of all around, the model accuracy will be obviously declining with the increase of the amount of data. This may because there are so much pseudo data, so the optimization for identification of true and fake data has become a main job, and optimization for forecast of output has become secondary work. After adjusting the superparameters for several times, the average results of ten tests are shown in row 11 of Table 4. It can be seen that the accuracy of the model has been further improved by using this method.

5.4 Analysis of model adjustment results

Through the adjustment and improvement of the structure, output range and training methods of DNN, this paper finally builds the DRN-GAN model. The MSE losses of the model reach the order of 1e-08, the R^2 coefficients reach more than 0.999, and the average coefficients are higher than other models shown in Table 4. Comparison of test results of all the improved model is shown in rows 8 through 11 of Table 4. With the continuous improvement and

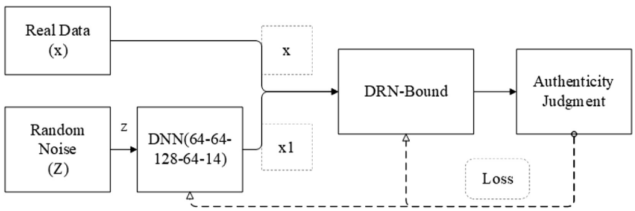


Figure 13.
DRN-GAN

adjustment of the model, the accuracy of the model is continuously improved, which proves that the above improvement schemes have produced positive effects.

6. Comparison and analysis of different CBM models

Through the modeling and testing of traditional models and deep learning models of CBM, this paper obtains the evaluation coefficients of each model under the UCI CBM dataset. As can be seen from the test results in Sections 4 and 5, there are obvious differences in the test effects of different models on the dataset, which is determined by the characteristics of the models themselves and related to the adaptability of the models to the dataset. The precision of each model in the test set is shown in Table 4. The best data in each category are shown in italic.

We can see that in traditional models, the effect of HMM model on this dataset is not ideal. This may be related to the model itself which is difficult to extract information from small dataset. SVR model is suitable for fitting the task under small dataset and obtain good effect on this dataset. In particular, in view of the GTC decay, SVR model fitting is very accurate but decrease for GT decay.

In the deep learning models, the fitting effect of Adaboost is not ideal, which may be related to the difficulty of updating the data distribution due to the limited small dataset. Both DNN model and RandomForest model have good fitting effects on the dataset. Meanwhile, the improvement and adjustment of DNN model also significantly improve the fitting accuracy. The final improved model, DRN-GAN, has a better prediction accuracy for GT decay, and its prediction of GTC decay is only slightly worse than SVR. Meanwhile, the average MSEloss and R^2 coefficient are significantly higher than other models. To sum up, among several CBM methods, DRN-GAN model has a better prediction effect on the dataset.

7. Conclusion

System maintenance behaviors ensure the safe operation of large mechanical systems to a certain extent, which not only ensure the economic interests of the system owners but also avoid the occurrence of accidental safety accidents to a great extent. CBM methods are based on historical data modeling, which can continuously monitor system performance without interrupting system operation and accurately predict equipment performance decay coefficients, thus providing important reference information for replacement and maintenance work.

This paper has mainly completed the following work based on the UCI CBM Data:

(1) The data processing process is completed on the dataset, and the purpose of shortening the training time and improving the training effect is achieved through feature clipping, data preprocessing and enhancement.

(2) The modeling and testing of traditional CBM methods (represented by SVR and HMM), and CBM methods based on deep learning (represented by Adaboost, RandomForest and DNN) are carried out on the dataset. The prediction accuracy and algorithm performance reference of these CBM methods in the prediction of GT equipment performance decay parameters are obtained.

(3) The DRN-GAN model is obtained. The simulation results show that the comprehensive performance and accuracy of DRN-GAN is a better model than other representative models mentioned in this paper, and the Aveloss coefficient reaches 6.80e-08 and R2-Ave coefficient reaches 0.999415. These coefficients are better than other models shown in this paper.

The main innovative achievements of this paper are as follows:

(1) The main applicable CBM algorithms are modeled and tested on the UCI CBM dataset, and the performance and accuracy of each algorithm on the dataset are evaluated. This process provides a reference for the modeling and improvement of a new high-precision model: DRN-GAN.

(2) In the process of building and improving the new high-precision model based on DNN algorithm, this paper innovatively uses the main framework of 14-64-64-16-2/1, residual structure, restrictions on the output range and data expansion training method used in ACGAN to construct the DRN-GAN model. According to the test results, the model has a higher average accuracy and a better effect on the dataset, and its Aveloss coefficient reaches 6.80e-08 and R2-Ave coefficient reaches 0.999415.

(3) In this paper, the test accuracy of each model and the performance on the dataset are comprehensively compared and analyzed. The simulation results show that the proposed DRN-GAN model has a better prediction effect for the maintenance of the GT of the naval propulsion system.

In addition, there is still room for further research in this paper. This paper only considers and models the main deep learning algorithms at present. With the development of deep learning, various new methods will be proposed constantly. Once they are proved to be effective, applying them to the problems of GT maintenance may further improve the prediction effect. At the same time, if additional sensor information can be obtained, such as infrared image of equipment and vibration frequency etc., the model may be more accurate in predicting the performance decay coefficients.

References

Ahmad, R. and Kamaruddin, S. (2012), “An overview of time-based and condition-based maintenance in industrial application”, *Computers and Industrial Engineering*, Vol. 63, pp. 135-149.

Asuncion, A. and Newman, D. (2007), *UCI Machine Learning Repository*.

Bagavathiappan, S., Lahiri, B., Saravanan, T., Philip, J. and Jayakumar, T. (2013), “Infrared thermography for condition monitoring–a review”, *Infrared Physics and Technology*, Vol. 60, pp. 35-55.

Basurko, O.C. and Uriondo, Z. (2015), “Condition-based maintenance for medium speed diesel engines used in vessels in operation”, *Applied Thermal Engineering*, Vol. 80, pp. 404-412.

Berry, M.J. and Linoff, G.S. (2004), *Data Mining Techniques: For Marketing, Sales, and Customer Relationship Management*, John Wiley & Sons, Indianapolis, IN.

Borisov, K., Calvert, T.E., Kleppe, J.A., Martin, E. and Trzynadlowski, A.M. (2006), “Experimental investigation of a naval propulsion drive model with the PWM-based attenuation of the acoustic and electromagnetic noise”, *IEEE Transactions on Industrial Electronics*, Vol. 53, pp. 450-457.

Breiman, L. (1996), “Bagging predictors”, *Machine Learning*, Vol. 24, pp. 123-140.

Breiman, L. (2001), “Random forests”, *Machine Learning*, Vol. 45, pp. 5-32.

Burdack, J., Horst, F., Giesselbach, S., Hassan, I., Daffner, S. and Schöllhorn, W.I. (2020), “Systematic comparison of the influence of different data preprocessing methods on the performance of gait classifications using machine learning”, *Frontiers in Bioengineering and Biotechnology*, Vol. 8, p. 260.

Carey, B., Dan, M. and Tarik, A.A. (2000), “Condition-based maintenance of machines using hidden Markov models”, *Mechanical Systems and Signal Processing*, Vol. 14, pp. 597-612.

Castro, I.T., Basten, R.J. and Van Houtum, G.J. (2020), "Maintenance cost evaluation for heterogeneous complex systems under continuous monitoring", *Reliability Engineering and System Safety*, Vol. 200, p. 106745.

Chang, C.C. (2014), "Optimum preventive maintenance policies for systems subject to random working times, replacement, and minimal repair", *Computers and Industrial Engineering*, Vol. 67, pp. 185-194.

Cipollini, F., Oneto, L., Coraddu, A., Murphy, A.J. and Anguita, D. (2018), "Condition-based maintenance of naval propulsion systems: data analysis with minimal feedback", *Reliability Engineering and System Safety*, Vol. 177, pp. 12-23.

Cipollini, F., Oneto, L., Coraddu, A., Murphy, A.J. and Anguita, D. (2018), "Condition-based maintenance of naval propulsion systems with supervised data analysis", *Ocean Engineering*, Vol. 149, pp. 268-278.

Coraddu, A., Oneto, L., Ghio, A., Savio, S., Anguita, D. and Figari, M. (2016), "Machine learning approaches for improving condition-based maintenance of naval propulsion plants", *Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment*, Vol. 230, pp. 136-153.

Dhumale, R. (2018), "An overview of artificial neural networks: part 3 activation functions", *Artificial Intelligent Systems and Machine Learning*, Vol. 10, pp. 66-71.

Doerry, N., Amy, J. and Krolick, C.. (2015), "History and the status of electric ship propulsion, integrated power systems, and future trends in the US navy", *Proceedings of the IEEE*, Vol. 103, pp. 2243-2251.

Freund, Y. and Schapire, R.E. (1997), "A decision-theoretic generalization of on-line learning and an application to boosting", *Journal of Computer and System Sciences*, Vol. 55, pp. 119-139.

Glorot, X., Bordes, A. and Bengio, Y. (2011), "Deep sparse rectifier neural networks", *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, JMLR Workshop and Conference Proceedings*, pp. 315-323.

Hanson, R.K. (2011), "Applications of quantitative laser sensors to kinetics, propulsion and practical energy systems", *Proceedings of the Combustion Institute*, Vol. 33, pp. 1-40.

He, K., Zhang, X., Ren, S. and Sun, J. (2016), "Deep residual learning for image recognition", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770-778.

Jardine, A.K., Lin, D. and Banjevic, D. (2006), "A review on machinery diagnostics and prognostics implementing condition-based maintenance", *Mechanical Systems and Signal Processing*, Vol. 20, pp. 1483-1510.

Kobbacy, K.A.H. and Murthy, D.N.P. (2008), *Complex System Maintenance Handbook*, Springer, London.

Koduri, S.B., Gunisetti, L., Ramesh, C.R., Mutyalu, K. and Ganesh, D. (2019), "Prediction of crop production using adaboost regression method", *Journal of Physics: Conference Series*, IOP Publishing, Andhra Pradesh, p. 012005.

Kothamasu, R. and Huang, S.H. (2007), "Adaptive Mamdani fuzzy model for condition-based maintenance", *Fuzzy Sets and Systems*, Vol. 158, pp. 2715-2733.

Krizhevsky, A., Sutskever, I. and Hinton, G.E. (2012), "Imagenet classification with deep convolutional neural networks", *Advances in Neural Information Processing Systems*, Vol. 25, pp. 1097-1105.

Lazakis, I., Dikis, K., Michala, A.L. and Theotokatos, G. (2016), "Advanced ship systems condition monitoring for enhanced inspection, maintenance and decision making in ship operations", *Transportation Research Procedia*, Vol. 14, pp. 1679-1688.

Manikandan, S. and Duraivelu, K. (2021), "Fault diagnosis of various rotating equipment using machine learning approaches–a review", *Proceedings of the Institution of Mechanical Engineers, Part E: Journal of Process Mechanical Engineering*, Vol, *235*, pp. 629-642.

Martelli, M. and Figari, M. (2017), "Real-time model-based design for codlag propulsion control strategies", *Ocean Engineering*, Vol. 141, pp. 265-276.

Odena, A., Olah, C. and Shlens, J. (2017), "Conditional image synthesis with auxiliary classifier gans", *International Conference on Machine Learning*, PMLR, Sydney, pp. 2642-2651.

Özgür-Ünlüakın, D., Türkali, B. and Aksezer, S.Ç. (2021), "Cost-effective fault diagnosis of a multi-component dynamic system under corrective maintenance", *Applied Soft Computing*, Vol. 102, p. 107092.

Pal, P., Datta, R., Segev, A. and Yasinsac, A. (2019), "Condition based maintenance of turbine and compressor of a codlag naval propulsion system using deep neural network", *6th International Conference on Artificial Intelligence and Applications (AIAP-2019)*.

Płaczek, S. and Płaczek, A. (2018), "Learning algorithm analysis for deep neural network with relu activation functions", *ITM Web of Conferences*, EDP Sciences, Les Ulis, France, p. 01009.

Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A. and Chen, X. (2016), "Improved techniques for training gans", *Advances in Neural Information Processing Systems*, Vol. 29, pp. 2234-2242.

Selvik, J.T. and Aven, T. (2011), "A framework for reliability and risk centered maintenance", *Reliability Engineering and System Safety*, Vol. 96, pp. 324-331.

Simani, S., Fantuzzi, C. and Patton, R.J. (2003), "Model-based fault diagnosis techniques", *Model-based Fault Diagnosis in Dynamic Systems Using Identification Techniques*, Springer, London, pp. 19-60.

Smola, A.J. and Schölkopf, B. (2004), "A tutorial on support vector regression", *Statistics and Computing*, Vol. 14, pp. 199-222.

Tan, Y., Niu, C., Tian, H., Hou, L. and Zhang, J. (2019), "A one-class SVM based approach for condition-based maintenance of a naval propulsion plant with limited labeled data", *Ocean Engineering*, Vol. 193, p. 106592.

Wang, S., Deng, Z., Chung, F.L. and Hu, W. (2013), "From Gaussian kernel density estimation to kernel methods", *International Journal of Machine Learning and Cybernetics*, Vol. 4, pp. 119-137.

Widodo, A. and Yang, B.S. (2007), "Support vector machine in machine condition monitoring and fault diagnosis", *Mechanical Systems and Signal Processing*, Vol. 21, pp. 2560-2574.

Corresponding author
Jin Cui can be contacted at: jincui@buaa.edu.cn