Intelligent Fault Diagnosis of Machinery Using Digital Twin-assisted Deep Transfer Learning

Min Xia[1], Haidong Shao[2*], Darren Williams[3], Siliang Lu[4], Lei Shu[5], Clarence W. de Silva[6]

1. Department of Engineering, Lancaster University, Lancaster, LA1 4YW, United Kingdom
2. State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body, College of Mechanical and Vehicle Engineering, Hunan University, Changsha 410082, China
3. The Welding Institute, Cambridge, CB21 6AL, United Kingdom
4. College of Electrical Engineering and Automation, Anhui University, Hefei, 230601, China
5. College of Engineering, Nanjing Agricultural University, Nanjing, 210095, China.
6. Department of Mechanical Engineering, The University of British Columbia, Vancouver, V6T 1Z4, Canada

Corresponding author: hdshao@hnu.edu.cn (Haidong Shao)

Abstract: Digital twin (DT) is emerging as a key technology for smart manufacturing. The high fidelity DT model of the physical assets can produce system performance data that is close to reality, which provides remarkable opportunities for machine fault diagnosis when the measured fault condition data are insufficient. This paper presents an intelligent fault diagnosis framework for machinery based on DT and deep transfer learning. First, the DT model of the machine is built by establishing the simulation model and with further updating through continuously measured data from the physical asset. Second, all important machine conditions can be simulated from the built DT. Third, a new-type deep structure based on novel sparse de-noising auto-encoder (NSDAE) is developed and pre-trained with condition data from the source domain, as generated from the DT. Then, to achieve accurate machine fault diagnosis with possible variations in working conditions and system characteristics, the pre-trained NSDAE is fine-tuned using parameter transfer with only one sample from the target domain. The presented method is validated through a case study of triplex pump fault diagnosis. The experimental results demonstrate that the proposed method achieves intelligent fault diagnosis with a limited amount of measured data and outperforms other state-of-the-art data-driven methods.

Keywords: Digital twin; Fault diagnosis; Novel sparse de-noising auto-encoder; Deep transfer learning.

1: Introduction

With the emerging information and communication technologies, the manufacturing sector is transforming rapidly into next-generation smart manufacturing with enhanced modularity, flexibility, and reliability [1][2]. The availability of a manufacturing system is crucial in achieving high productivity in the competitive market. To reduce system downtime and unexpected failure of increasingly complex industrial machines and instruments, machine fault detection and diagnosis play a significant role [3]. Compared with the traditional fault diagnosis where the relationship between the monitored data and the health states of machines comes from the abundant human expertise of engineers, intelligent fault diagnosis (IFD) applies machine learning theories to machine fault diagnosis, which automates the process of fault detection and classification [4]. The traditional IFD methods contain three steps: sensor data acquisition, feature extraction, and fault classification [5]. Feature extraction involves extracting the features from the collected data through different domain analysis activities such as those in time domain, frequency domain, and time-frequency domain [6]. In the final fault classification step, the extracted features are used to train machine learning models such as artificial neural networks, support vector machine, and random forest, to carry out fault prediction [7]. However, the manual extraction of features has significant drawbacks as it is task intensive, with different fault classification tasks and varying machine working conditions.

Deep learning (DL) has been a significant breakthrough in the field of artificial intelligence in recent decades, which greatly overcomes the shortcomings in the traditional IFD methods. Due to the capability of automated extraction of representative features from raw data and accurately establishing nonlinear mapping of different health conditions, popular DL techniques including deep belief network (DBN) [8],

stacked auto-encoder (SAE) [9], and convolutional neural network (CNN) [10] have attracted increasing attention and become the very popular tools for IFD. Shao et al. [11] developed a novel method for IFD of roller bearings using deep wavelet auto-encoder and extreme learning machine, which achieved more efficient diagnosis performance than traditional machine learning approaches and standard deep learning methods. Abid et al. [12] proposed a new DL architecture called deep-SincNet for a multi-fault diagnosis task. High accuracy for several separated and combined faults, more physical interpretability, high robustness against noisy environments, and a significant gain in implementation cost were achieved by their approach. However, training a DL model from scratch for IFD will need a large amount of training data that contain all possible machine fault conditions. This is a major challenge in practical applications since the measured data only for some fault conditions can be far insufficient.

To solve the issue of limited training data, transfer learning (TL) has been introduced, which has shown remarkable capability in obtaining a satisfactory deep architecture by fine-tuning a DL model that has been pre-trained in other tasks [13-15]. Recently, TL has been also utilized in IFD. Wen et al. [16] presented a transfer CNN for fault diagnosis based on ResNet-50, which achieved accurate diagnosis results on bearing faults and self-priming centrifugal pump faults. Zhu et al. [17] improved the CNN-based TL method by calculating the domain loss by a linear combination of multiple Gaussian kernels, which enhanced the ability of adaptation compared to using a single kernel. Shao et al. [18] developed a fault diagnosis approach with modified TL by introducing stochastic pooling and Leaky rectified linear unit, which was successfully applied to analyze thermal images of a rotor-bearing system under different working conditions. However, the pre-trained models from completely different applications, on which TL is based, may not have learned the most representative features. Thus, the robustness of the diagnosis performance can be significantly impacted in practical scenarios where the

working conditions and system characteristics are not fixed. The parameter transfer learning strategy achieves more robust fault diagnosis results where the pre-trained model that is trained on data collected from the original system is fine-tuned using a small amount of samples in the target domain. However, the lack of measured data for all fault conditions in the source domain remains a major bottleneck in using parameter transfer learning [19].

With the advances in multi-physics modeling and simulation, advanced sensing and signal processing, cloud and edge computing, and 5G, the concept of digital twin (DT) has gained increasing attention in recent years in various areas of smart manufacturing such as product design [20], job shop scheduling [21], and process optimization [22]. A high fidelity digital model of the physical asset can produce simulation data of the system performance that is close to the real asset. DT provides new opportunities for IFD by generating simulation data for the unavailable fault conditions. Some researchers have recently investigated DT in fault diagnosis and predictive maintenance. Wang et.al [23] proposed a preliminary DT reference model for rotating machinery. A parameter sensitivity analysis-based model updating scheme was investigated to enhance the model adaptability. The developed DT model was used in diagnosing rotor unbalance fault and predicting its progression. Jain et al. [24] developed a DT that estimated the measurable characteristic outputs of a photo-voltaic energy conversion unit (PVECU) in real-time. The model generated an error residual for various typical faults that were used for fault diagnosis purposes. However, no systematic approach has been developed yet for improving the machine fault diagnosis performance using DT, to overcome the practical issue of limited measured data.

The present paper develops a new DT-assisted deep transfer learning framework for machine IFD. The DT model of the machine provides the simulation data of the possible fault conditions, and hence overcomes the problem of data unavailability for some fault conditions. The challenge of varying system

characteristics is addressed by adaptively updating the DT model. The improved deep transfer learning method based on sparse de-noising auto-encoder can fully utilize the trained model with source domain data and a very limited amount of data in the target domain. The performance of the proposed IFD is evaluated through a case study of triplex pump fault diagnosis. The main contributions of the paper are as follows:

1) A new framework for machine fault diagnosis using DT and deep transfer learning is proposed, which solves the problem of limited or unavailable measured fault condition data of machines with varying working conditions or shifted system characteristics. The DT of the machine is built and continuously updated to generate possible fault conditions close to those of the actual asset. The produced fault condition data from the DT model constructs the training data in the source domain for transfer learning.
2) A NSDAE is developed to construct the deep structure. The Swish activation function is used, which achieves superior results compared with the commonly used rectified linear unit (ReLU) or other activation functions.
3) For the training of NSDA, the maximum correntropy (Max-corr) is introduced in the cost function. Compared with mean square error (MSE), Max-corr is more effective for measuring the local similarity of two complex signals.
4) With just one sample in the target-domain, the proposed method achieves one-shot learning with the successful transfer of the model parameters of the pre-trained source-domain NSDAE to the target-domain NSDAE, which solves the difficulties of effective training using limited data in the target-domain.

The organization of the rest of the paper is as follows. In Section 2, the sparse de-noising

auto-encoder is introduced. Section 3presents the details of the proposed method and its implementation. The experimental study, evaluation of the collected results, and discussion are presented in Section 4. Section 5 concludes this paper and highlights the planned future work.

2. The principle of de-noising auto-encoder

Despite DBN has the capability of providing joint probability distribution for the input data, which is hard to analyze the real-valued vibration data in the IFD tasks. Due to robust supervised feature learning ability, CNN is treated as the most widely applied deep learning model in IFD field, however, CNN has complex structure and high computational cost. As a completely unsupervised learning model, SAE is trained more easily and effectively than DBN and CNN, which has derived many improved forms, such as sparse auto-encoder and de-noising auto-encoder. Sparse de-noising auto-encoder (SDAE) fully combines the advantages of sparse auto-encoder and de-noising auto-encoder, which can effectively learn the sparse feature representation from noisy input samples. **Fig. 1** shows the model architecture of an SDAE. The related main formulas are listed below.

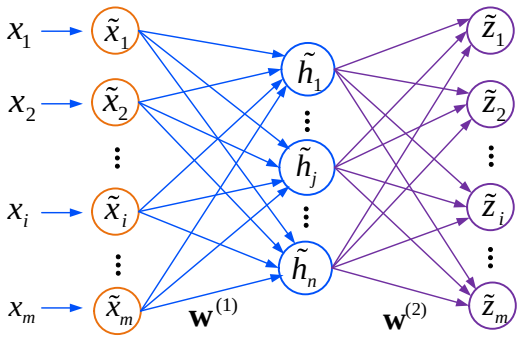


Fig. 1. The model architecture of an SDAE.

$$\tilde{\mathbf{x}} = \mathbf{x} + \boldsymbol{N}(0, \delta^2\mathbf{I}) \tag{1}$$

$$\tilde{\boldsymbol{h}} = f_H(\mathbf{w}^{(1)}\tilde{\boldsymbol{x}} + \mathbf{b}^{(1)}) \tag{2}$$

$$\tilde{\boldsymbol{z}} = f_O(\mathbf{w}^{(2)}\tilde{\boldsymbol{h}} + \mathbf{b}^{(2)}) \tag{3}$$

$$C_S = l_1 + l_2 + l_3 \tag{4}$$

with

$$l_1 = \frac{1}{2}\sum_{i=1}^{m}(\tilde{z}_i - x_i)^2 \tag{5}$$

$$l_2 = \beta\left(\sum_{j=1}^{n} r \log\frac{r}{\hat{r}_j} + (1-r)\log\frac{1-r}{1-\hat{r}_j}\right) \tag{6}$$

$$l_3 = \frac{\lambda}{2}\left(\sum_{i=1}^{m}\sum_{j=1}^{n}\left((w_{ji}^{(1)})^2 + (w_{ji}^{(2)})^2\right)\right) \tag{7}$$

in which $\boldsymbol{x} = [x_1, x_2, \mathrm{L}\ , x_m]$ represents the m-dimensional true sample with label; $\tilde{\boldsymbol{x}} = [\tilde{x}_1, \tilde{x}_2, \mathrm{L}\ , \tilde{x}_m]$ represents the noisy sample; $\boldsymbol{N}(0, \delta^2\mathbf{I})$ represents the Gaussian noise with a noise level of δ, f_H and f_O are the activation functions of the hidden layer and output layer, respectively; $(\mathbf{w}^{(1)}, \mathbf{b}^{(1)})$ and $(\mathbf{w}^{(2)}, \mathbf{b}^{(2)})$ are (weight, bias) connecting the input and the hidden layers, the hidden and output layers, respectively; $\tilde{\boldsymbol{h}} = [\tilde{h}_1, \tilde{h}_2, \mathrm{L}\ , \tilde{h}_n]$ and $\tilde{\boldsymbol{z}} = [\tilde{z}_1, \tilde{z}_2, \mathrm{L}\ , \tilde{z}_m]$ are n-dimensional feature vector and m-dimensional reconstruction vector of $\tilde{\boldsymbol{x}}$; C_S represents the cost function of SDAE consisting of MSE l_1, sparse penalty term l_2 and weight decay term l_3; β, r and λ are sparse penalty factor, sparse constant and weight decay coefficient, respectively; and $w_{ji}^{(1)} \in \mathbf{w}^{(1)}$ and $w_{ji}^{(2)} \in \mathbf{w}^{(2)}$ are connection weights between the i^{th} input unit and j^{th} hidden unit, and the j^{th} hidden unit and i^{th} output unit, respectively.

3. The proposed method

Based on the DT technique and deep transfer learning model, this paper proposes a new intelligent fault diagnosis framework for machinery. The proposed deep transfer learning model mainly contains three parts: NSDAE construction, parameter transfer of the stacked NSDAE, and DT-assisted deep transfer learning for IFD.

3.1 NSDAE construction

In this section, NSDAE is constructed with Swish and maximum correntropy (Max-corr). ReLU is the most widely applied activation function in designing deep neural networks. However, ReLU outputs zero on negative input, which limits its learning capability. The new type of activation function developed by Google Brain in 2017, the Swish function, simultaneously possesses the characteristics of bounded

below, unbounded above, smooth and non-monotonic. The comparative results based on several benchmark datasets have demonstrated the superiority of Swish over ReLU and other popular activation functions [25].

Fig. 2 shows the waveform of the Swish activation function. The mathematical expression for a noisy sample x_i is defined as follows.

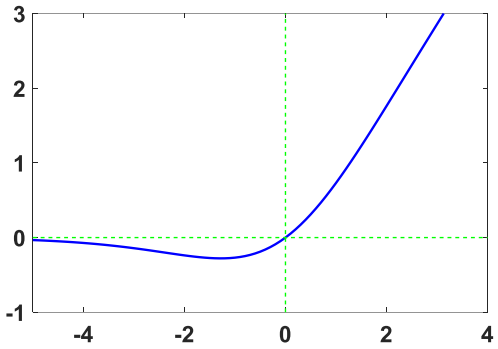


Fig. 2. The waveform of the Swish activation function.

$$Swish(x_i) = \frac{x_i}{1+\exp(-x_i)} \tag{8}$$

Based on the Swish activation function, the hidden output h_j and the reconstruction output z_i of input units x_i can be computed as

$$h_j = \frac{\left(\sum_{i=1}^{m} w_{ji}^{(1)} x_i + b_j^{(1)}\right)}{1+\exp\left(\sum_{i=1}^{m} w_{ji}^{(1)} x_i + b_j^{(1)}\right)} \tag{9}$$

$$z_i = \varphi\left(\sum_{j=1}^{n} \frac{w_{ji}^{(2)}\left(\sum_{i=1}^{m} w_{ji}^{(1)} x_i + b_j^{(1)}\right)}{1+\exp\left(\sum_{i=1}^{m} w_{ji}^{(1)} x_i + b_j^{(1)}\right)} + b_i^{(2)}\right) \tag{10}$$

where $b_j^{(1)} \in \mathbf{b}^{(1)}$ and $b_i^{(2)} \in \mathbf{b}^{(2)}$ represent biases of h_j and z_i, respectively, and $\varphi(\cdot)$ represents the sigmoid function. In the present paper, considering that all of the inputs are normalized to the range [0, 1], thus, sigmoid activation function is selected in the output layer to well make the reconstructed output get close to the original input.

To effectively measure the local similarity between the true input sample and the reconstruction sample, Max-corr is used here, which has shown better performance than MSE. Then, the NSDAE can be constructed, whose parameter sets $\mathbf{w} = \{w_{ji}^{(1)}, w_{ji}^{(2)}\}$ and $\mathbf{b} = \{b_j^{(1)}, b_i^{(2)}\}$ are adjusted based on the

gradient descent algorithm with the learning rate decay and the momentum term as

$$\mathbf{w}_{q+1} = \mathbf{w}_q - \xi_q(\partial C_{\mathrm{N}}/\mathbf{w}_q) + \varepsilon(\mathbf{w}_q - \mathbf{w}_{q-1}) \tag{11}$$

$$\mathbf{b}_{q+1} = \mathbf{b}_q - \xi_q(\partial C_{\mathrm{N}}/\mathbf{b}_q) + \varepsilon(\mathbf{b}_q - \mathbf{b}_{q-1}) \tag{12}$$

$$C_{\mathrm{N}} = -\mathrm{l}_4 + \mathrm{l}_2 + \mathrm{l}_3 \tag{13}$$

$$\mathrm{l}_4 = \frac{1}{\sqrt{2\pi}\tau}\sum_{i=1}^{m}\exp\left(-\frac{(\%_i - x_i)^2}{2\tau^2}\right) \tag{14}$$

$$\xi_{q+1} = \xi_q / \rho \tag{15}$$

where q represents the current iteration number; C_{N} is the cost function of NSDAE; l_4 is the Max-corr, which is more effective in measuring the local similarity of two complex signals than MSE; τ is the kernel width adjustable parameter; ξ_q is the current learning rate; ρ is a decay factor; and ε is momentum factor. Based on multiple NSDAEs with a softmax classifier, a stacked NSDAE can be built to learn high-level features.

3.2 Parameter transfer of the stacked NSDAE

Parameter transfer learning is an effective strategy to greatly improve the training efficiency of a stacked NSDAE that has already been pre-trained. With parameter transfer learning, the stacked NSDAE trained with data samples from one domain (source domain) can well analyze the data samples from another domain (target domain) with different distributions. In the present paper, the transferable parameters consist of all the hyperparameters, weights, and biases in different layers. Among them, all of the weights and biases are pre-trained and then further fine-tuned. The initial values of the four transferable parameters for the target-domain stacked NSDAE are provided by the source-domain stacked NSDAE that has been pre-trained. **Fig. 3** shows the schematic diagram of how to introduce the parameter transfer learning into the training and testing of the stacked NSDAE with three NSDAEs.

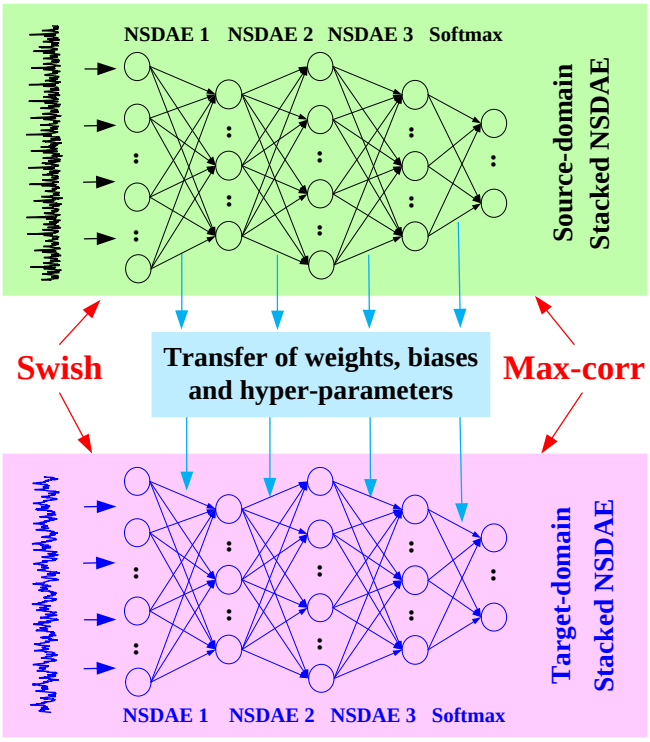


Fig. 3. Schematic diagram of parameter transfer for the stacked NSDAE.

3.3 DT-assisted deep transfer learning for IFD

The overall framework of the proposed DT-assisted deep transfer learning for machine fault diagnosis is shown in **Fig. 4**. A simulation model is built and updated with the measured system response to form the DT of the physical asset. The adaptively updated DT model is used to generate simulation data of all interested fault conditions of the machine that is used as the training data in the source domain. Then, the stacked NSDAE model with improved activation and cost function is constructed and pre-trained with data from the source domain. With only one sample from the conditions of the running machine under variable operating conditions and possible shifted system characteristics (the target domain), parameter transfer learning is carried out. The model after transfer learning can be used for effective machine fault diagnosis. The main steps of the proposed method are:

- **Step 1:** Create a simulation model of the actual machine (e.g., in MATLAB Simulink and Simscape).
- **Step 2:** Collect measured data of the system response and update the simulation model by

minimizing the difference of system responses between the simulation model and the measured data.

- **Step 3:** With the DT model established, simulate all interested faults of the machine. The simulation data generated from different machine conditions are treated as the source domain data.
- **Step 4:** Construct the stacked NSDAE with the Swish activation function and maximum correntropy.
- **Step 5:** The training and testing samples from the source domain are used to acquire a pre-trained stacked NSDAE with high and robust diagnosis accuracies.
- **Step 6:** To achieve machine fault diagnosis in the target domain (under varying working conditions or shifted system characteristics), the pre-trained stacked NSDAE is fine-tuned by only one training sample from the target domain to further adjust the model weights.
- **Step 7:** The testing samples from the target domain are used to check the effectiveness of the proposed method. The final stacked NSDAE can be deployed in the fault diagnosis of the running machine.

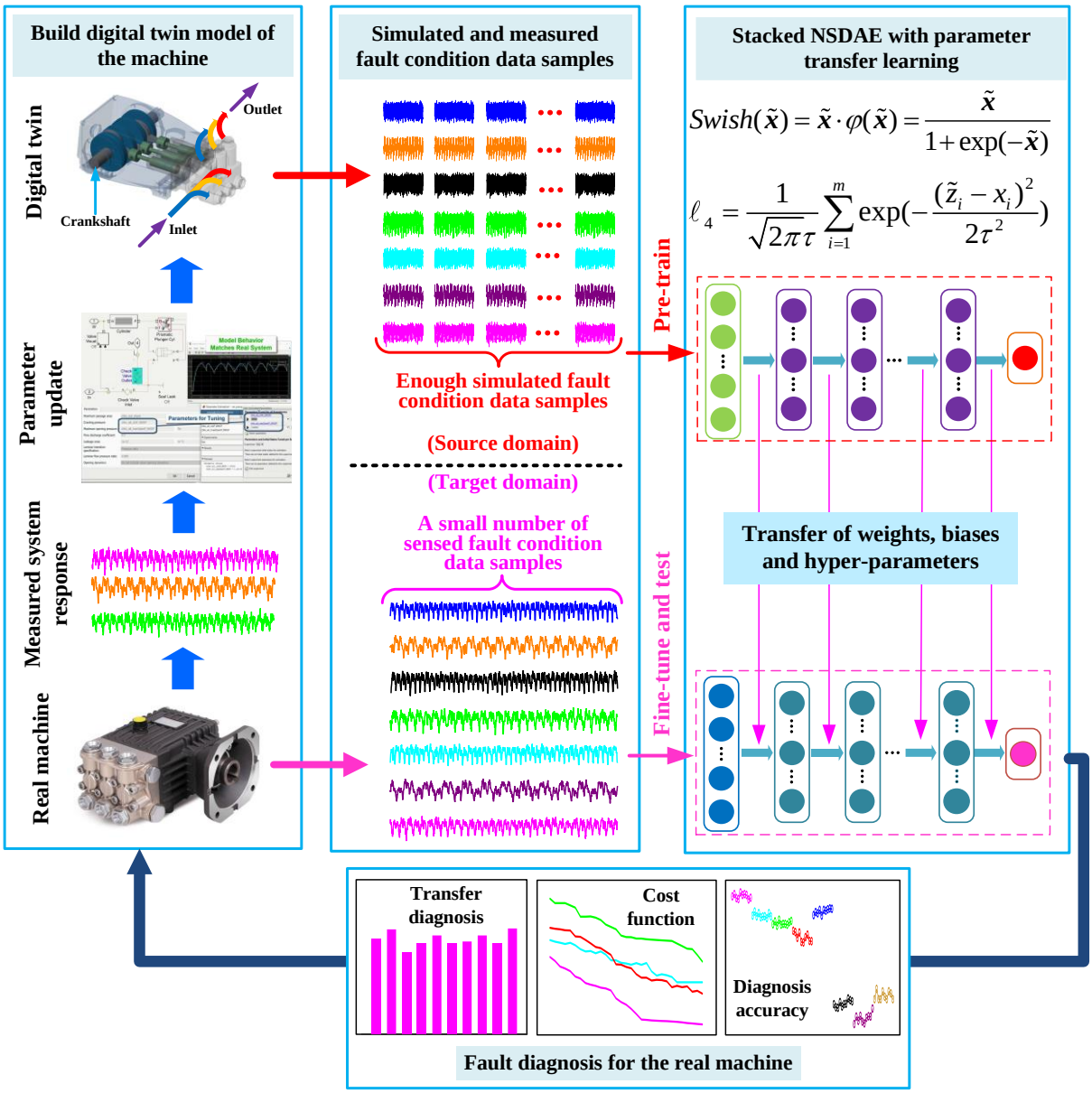


Fig. 4. The framework of the proposed DT-assisted deep transfer learning for IFD.

4. Case validation

In the present paper, the feasibility of the proposed DT-assisted machine IFD is evaluated through the fault diagnosis of a triplex pump, which is presented now.

4.1 Digital twin of the triplex pump

As shown in **Fig. 5**, a simulation model of an actual triplex pump is created using Simscape in Matlab from [26]. Here, we modify the system parameters including upper and lower pressures, at which levels the three check valves feeding the outlet will open and close, to simulate the shifted system

characteristics, e.g., changed temperature and property of the transported fluid. The system response of the model with modified parameters is then generated and used as measured data from the shifted real assets, to optimize the simulation model. In the present study, the DT model update is achieved through Simulink Design Optimization with automatic tuning of the model parameters. The gradient descent method with sequential quartic programming is applied to minimize the differences between the simulated and measured curves of outlet pressure.

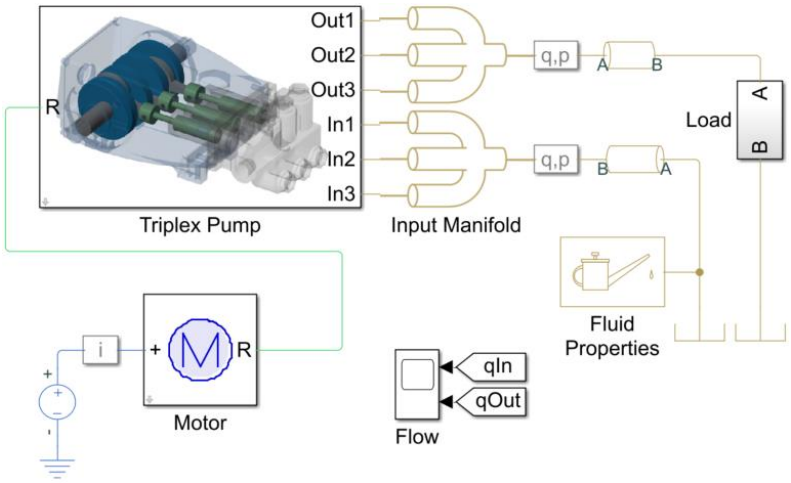


Fig. 5. Digital twin model of a triplex pump.

The pump DT is configured to simulate three typical types of pump faults including cylinder leak, blocked inlet, and increased bearing friction. These faults are parameterized as workspace variables and configured through the pump block dialog. In the present study, a total of seven machine conditions of the triplex pump are generated, including one health state, three types of single-fault states, and three types of compound-fault states.

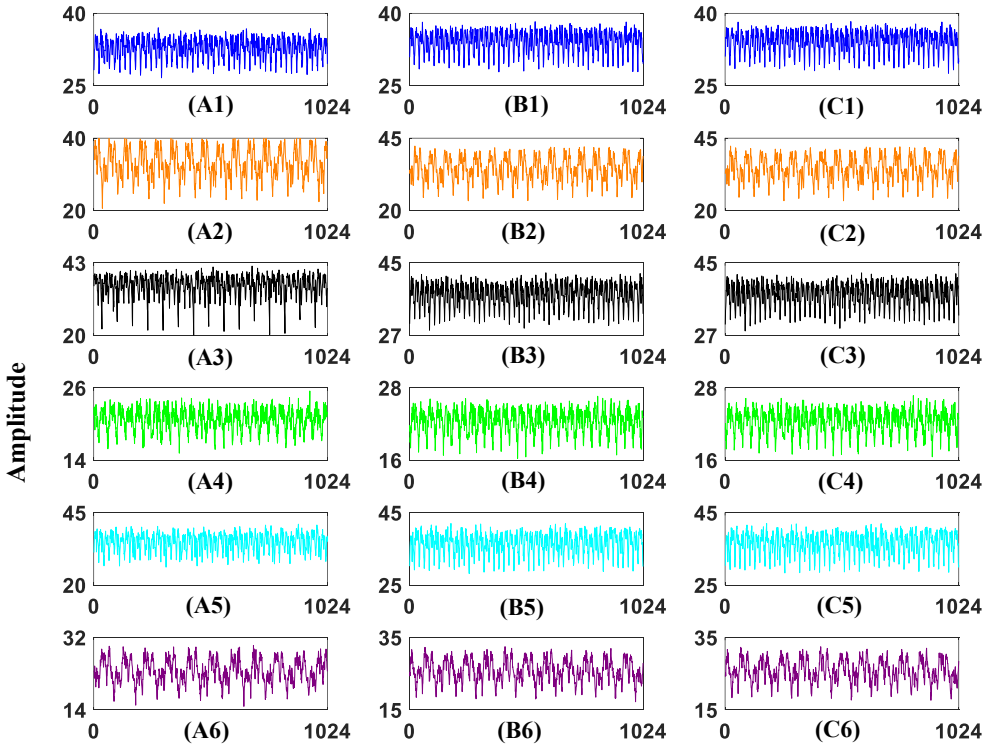
In the present study, three scenarios of simulation are run, and the collected datasets are marked as Dataset A, Dataset B, and Dataset C. Dataset A refers to the simulation data from the simulation model with original parameters. Dataset B refers to the actual condition data from the physical asset (in the present case study, we use simulation data from the simulation model with modified system parameters to

simulate the scenario of changing working conditions or system characteristics). Dataset C refers to the simulation data generated from the updated DT that has been optimized. The simulation duration for each run is 2.0s. The output flow data from the flow rate sensor is collected. Here, only the signals from 0.8s to 2s are selected to exclude the initial transient stage. For each of the seven machine conditions, 125 trails are run. The Datasets A, B, and C are listed in **Table 1**. Each sample includes 1200 data points (pump outlet flow data). The first samples of the three datasets are plotted in **Fig. 6**.

Table 1

The seven working conditions of triplex pumps

Working states of the triplex pump	Total number of the samples			Labels
	Data A	Data B	Data C	
Health	125	125	125	1
Leak	125	125	125	2
Blocking	125	125	125	3
Leak & Blocking	125	125	125	4
Bearing	125	125	125	5
Bearing & Leak	125	125	125	6
Bearing & Blocking	125	125	125	7



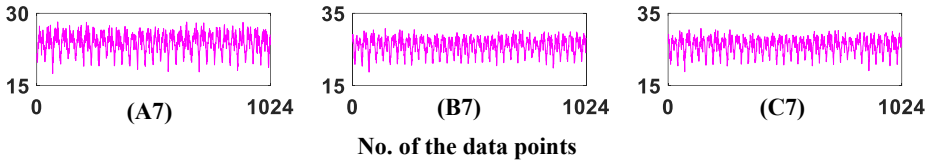


Fig. 6. Time-domain waveforms of the raw output flow: (A)-(C) represent Data A – Data C; (1)-(7) represent State 1 –State 7.

Four comparative tasks are designed to demonstrate the effectiveness of the proposed method in machine fault diagnosis, marked as Task 1, Task 2, Task 3, and Task 4. **Table 2** presents the strategies and objectives of the four tasks, and **Table 3** gives details on the sources and numbers of the training and testing samples in each task.

Table 2

Details about the strategies and objectives of the four tasks

Tasks	Strategies	Objectives
Task 1	Deep learning	Evaluate the superiority of the proposed stacked NSDAE in fault diagnosis using the raw output flow values
Task 2	Deep learning	Evaluate the diagnosis performance of the stacked NSDAE when the system characteristics change
Task 3	Deep learning	Evaluate the diagnosis performance with training data from the updated DT model only
Task 4	Deep transfer learning	Evaluate the diagnosis effectiveness of the proposed method with DT and one-shot parameter transfer learning

Table 3

Sources and numbers of the training and testing samples in each task

Tasks	Sources of training / testing samples	Numbers of training / testing samples
Task 1	Data A / Data A	75 / 50
Task 2	Data A / Data B	75 / 50
Task 3	Data C / Data B	75 / 50
Task 4	Data C / Data B	75 / 50

4.2 Results analysis in Task 1 and Task 2

Task 1 focuses on testing the diagnosis performance of the developed stacked NSDAE using the raw output flow data. The proposed method is compared with different deep learning methods including three types of stacked SDAEs, LeNet-5 CNN, Gaussian DBN, stacked LSTM, and stacked GRU. Also, three

types of shallow learning methods, SVM, RF, and ELM, are evaluated for comparison, where their inputs are three types of feature sets.

The training data is constructed using randomly selected 75 samples out of the 125 samples from Dataset A, while the remaining 50 samples are used as the testing data. To minimize the randomness of the diagnosis results, the proposed stacked NSDAE and the comparative methods are all carried out for ten runs, i.e., each method runs for ten independent runs using the random samples. **Fig. 7** shows the testing diagnosis results of the eight deep learning methods during the ten runs. Their average testing accuracies are given in **Table 4**, which are obtained by calculating the average value of the accuracies of ten runs. Specifically, the average accuracy of the proposed method as applied to 3500 (50*7*10) testing samples is 98.46% (3446/3500), compared with those for the contrastive methods are 96.11%, 95.20%, 93.51%, 96.63%, 85.49%, 83.71%, and 86.40%, respectively. Thus, it is verified that the proposed stacked NSDAE constructed with Swish and Max-corr has achieved the highest diagnosis accuracy and is more effective in fault diagnosis of the triplex pump than other popular deep learning methods.

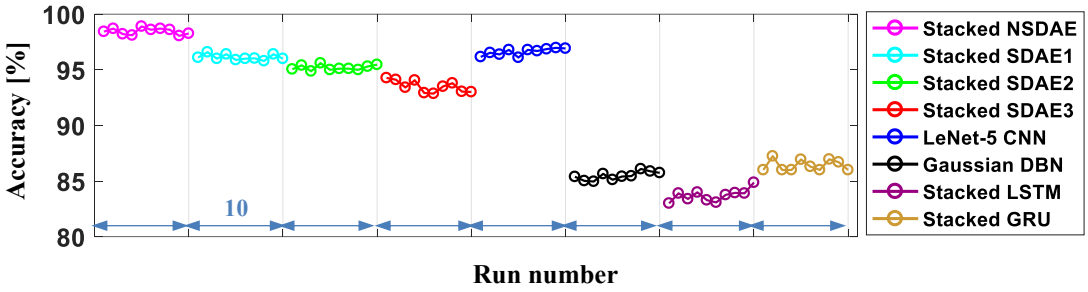


Fig. 7. The repeated diagnosis results based on the nine methods.

Table 4

Diagnosis results of the eight deep learning methods in Task 1

Diagnosis methods	Inputs	Average testing accuracies
Stacked NSDAE (Proposed)	**Raw data**	**98.46% (3446/3500)**
Stacked SDAE1 (Swish with CS)	Raw data	96.11% (3364/3500)
Stacked SDAE2 (ReLU with CN)	Raw data	95.20% (3332/3500)
Stacked SDAE3 (ReLU with CS)	Raw data	93.51% (3273/3500)

LeNet-5 CNN	Raw data	96.63% (3382/3500)
Gaussian DBN	Raw data	85.49% (2992/3500)
Stacked LSTM	Raw data	83.71% (2930/3500)
Stacked GRU	Raw data	86.40% (3024/3500)

Notes: C_S refers to the cost function of SDAE in Eq. (4) and C_N refers to the cost function of NSDAE in Eq. (13).

To further validate the advantages of Max-corr, the cost functions during the training are compared among the proposed stacked NSDAE and the other three types of stacked SDAEs. Consider the first base AEs as an example. The average cost function curves (from 30th iteration and 60th iteration) of the ten runs are plotted in **Fig. 8.** The results show that the training process of the proposed NSDAE converges faster than the other methods.

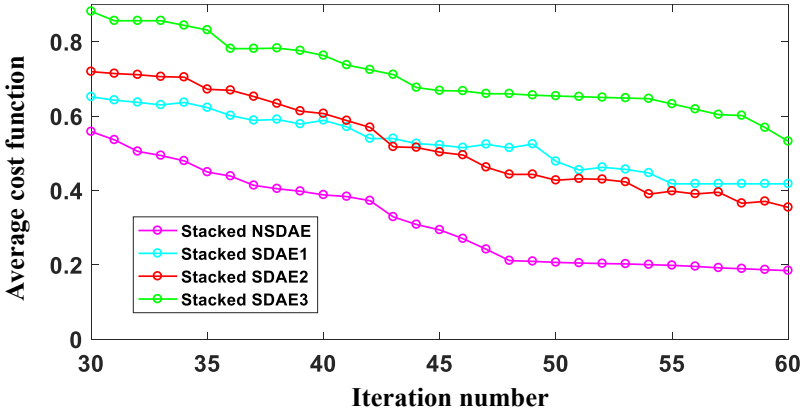


Fig. 8. The average cost function curves of the four types of stacked SDAEs.

For the training of shallow learning models, 14 features in total are extracted including 9 time-domain (TD) features (mean, variance, cumulative sum range, etc.) and 5 frequency-domain (FD) features (spectral kurtosis peak, peak magnitude in the power spectrum, etc.). The SVM, RF, and ELM models are trained using three types of feature sets. The average testing accuracies of the three shallow learning methods are listed in **Table 5**. It is found that the performances of these shallow learning methods are much below the performances of the deep learning methods. Also, the performances of shallow learning models highly depend on the selection of the input features, which presents difficulty in

generic applications.

Table 5

Diagnosis results of shallow learning methods in Task 1

Diagnosis methods	Inputs	Average testing accuracies
SVM	14 features (All)	78.34% (2742/3500)
SVM	9 TD features	76.14% (2665/3500)
SVM	5 FD features	81.17% (2841/3500)
RF	14 features (All)	80.26% (2809/3500)
RF	9 TD features	79.86% (2795/3500)
RF	5 FD features	83.11% (2909/3500)
ELM	14 features (All)	77.51% (2713/3500)
ELM	9 TD features	75.94% (2658/3500)
ELM	5 FD features	81.06% (2837/3500)

Task 2 is employed to test if the diagnosis performance of the stacked NSDAE and other deep learning methods would degrade when applied to the actual system (with simulation). 75 samples are randomly selected from the 125 samples in Dataset A for training. 50 samples from Dataset B are used as the testing samples. The detailed diagnosis results are given in **Table 6**. It is found that all of the studied methods trained with Dataset A fail to directly analyze Dataset B effectively. The main reason is that Dataset A from the initial model and Dataset B from the changed model have significant differences in distribution because of the system parameter shift. Therefore, the model update in the DT framework is crucial.

Table 6

Diagnosis results of different methods in Task 2

Diagnosis methods	Inputs	Average testing accuracies
Stacked NSDAE (Proposed)	Raw data	34.03% (1191/3500)
Stacked SDAE1 (Swish with CS)	Raw data	35.09% (1228/3500)
Stacked SDAE2 (ReLU with CN)	Raw data	28.51% (998/3500)
Stacked SDAE3 (ReLU with CS)	Raw data	33.17% (1161/3500)
LeNet-5 CNN	Raw data	36.71% (1285/3500)
Gaussian DBN	Raw data	26.89% (941/3500)
Stacked LSTM	Raw data	34.29% (1200/3500)
Stacked GRU	Raw data	32.26% (1129/3500)

SVM	5 FD features	38.11% (1344/3500)
RF	5 FD features	37.40% (1309/3500)
ELM	5 FD features	35.17% (1231/3500)

4.3 Results analysis in Task 3 and Task 4

Task 3 aims to evaluate that Dataset C (source domain) of the DT model has similar distribution with Dataset B (target domain) of the changed model after the DT model optimization. Different methods are trained with Dataset C and then directly applied to diagnose the different conditions in Dataset B. Similarly, randomly selected 75 out of 125 samples from Dataset C are adopted for training. 50 samples from Dataset B are used as the testing samples. Details of the diagnosis results are presented in **Table 7**. Compared with the results in Task 2, it can be seen that the diagnosis methods trained with Dataset C show a significant increase in diagnosis accuracy on Dataset B, which means the DT model is close to the simulated actual system. For the first run, the specific accuracy of the proposed stacked NSDAE is 75.71% (265 /350). The proposed method obtained better diagnosis results than the other approaches. The corresponding confusion matrix is plotted in **Fig. 9** and the *F*-measure values are shown in **Fig. 10**. The horizontal coordinate and vertical coordinate in **Fig. 9** successively represent the predicted and actual labels of state. The numbers on the main diagonal represent the diagnostic accuracy of each state, and the numbers located in other places represent the misdiagnosis rates.

Table 7
Diagnosis results of different methods in Task 3

Diagnosis methods	Inputs	Average testing accuracies
Stacked NSDAE (Proposed)	Raw data	76.06% (2662/3500)
Stacked SDAE1 (Swish with CS)	Raw data	71.17% (2491/3500)
Stacked SDAE2 (ReLU with CN)	Raw data	73.37% (2568/3500)
Stacked SDAE3 (ReLU with CS)	Raw data	69.20% (2422/3500)
LeNet-5 CNN	Raw data	73.83% (2584/3500)
Gaussian DBN	Raw data	66.80% (2338/3500)
Stacked LSTM	Raw data	63.17% (2211/3500)
Stacked GRU	Raw data	65.29% (2285/3500)
SVM	5 FD features	66.26% (2319/3500)

RF	5 FD features	68.54% (2399/3500)
ELM	5 FD features	65.63% (2297/3500)

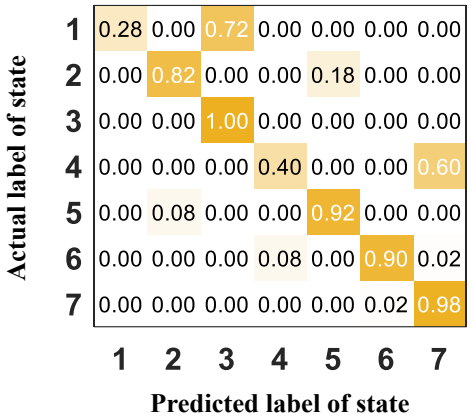


Fig. 9. Confusion matrix of the proposed stacked NSDAE for the first run.

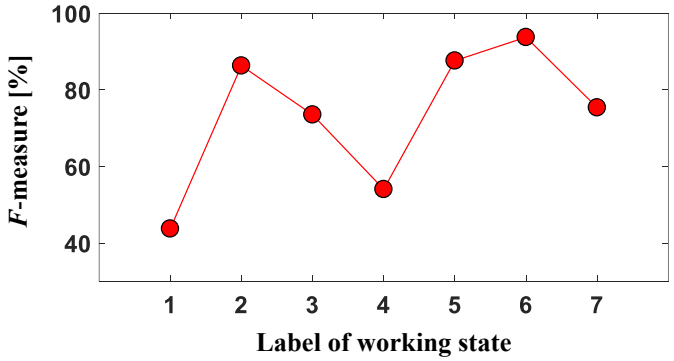


Fig. 10. *F*-measures of the proposed stacked NSDAE for different states in Task 3.

Task 4 verifies the performance of the proposed method with one-shot parameter transfer learning. Here, randomly selected 75 of 125 samples from Dataset C (source domain) are adopted as the training data in the source domain. After pre-training the stacked NSDAE, only one sample from Dataset B (target domain) is used to fine-tune the trained stacked NSDAE. Then, 50 samples from Dataset C are used for testing. Specifically, the total number of the source-domain training samples and target-domain training samples is 525 (75*7) and 7 (1*7), respectively. Each sample also contains 1200 sampling data points. After introducing parameter transfer learning, the diagnosis accuracies during the ten runs are shown in **Fig. 11** with an average value of 93.20% (3262/3500). Similarly, consider the first run as an example. The

testing accuracy is 93.14% (326 /350) and the *F*-measure values are shown in **Fig. 12**.

The comparison results between Tasks 3 and 4 show that DT and parameter transfer learning are promising tools to resolve the fault diagnosis problem when labeled fault condition samples are greatly insufficient. The DT of the triplex pump produced simulation data of all interested fault conditions, which are used as the training data in the source domain. With just one sample from the target domain, the constructed deep structure can achieve superior fault diagnosis results with parameter transfer learning. The hyper-parameters of the proposed stacked NSDAE are given in **Table 8**. Among them, the network structure is determined by experimentation and experience, i.e., the numbers of the neuron nodes in the input layer and final output layer are successively equal to the dimensions of the input data and output data; the number of the neuron nodes in the next hidden layer is about half that in the previous layer. The selection of other hyper-parameters is mainly based on grid search technique. Specifically, the grid search range of weight decay coefficient is between 0.001 and 0.009 by step of 0.001, the grid search range of sparse constant is between 0.1 and 0.9 by step of 0.1, and the grid search range of sparse penalty factor is between 1 and 9 by step of 1. The iteration number, initial learning rate, decay factor, and momentum are decided by experience as 60, 0.01, 1.1, and 0.8, respectively. After determining the hyper-parameters, the average computation time of the proposed deep transfer learning model per trial is about 238s under the following hardware configuration of Windows 10 64-bit operating system, Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz, and 16 GB RAM.

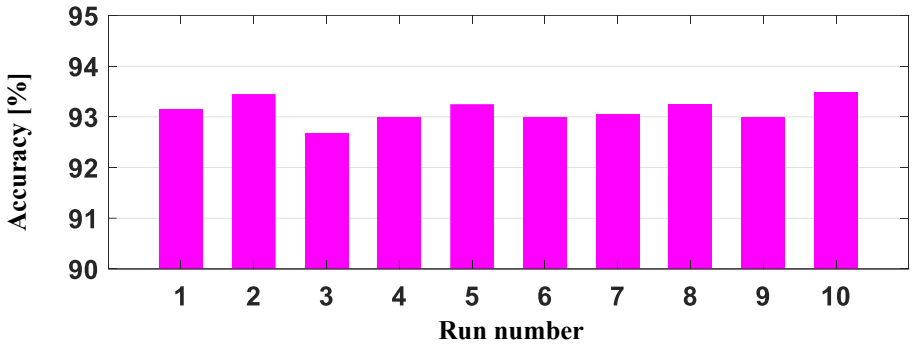


Fig. 11. The repeated diagnosis results based on the nine methods.

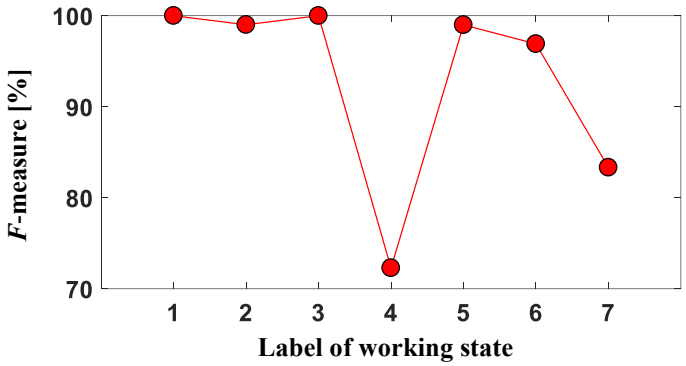


Fig. 12. *F*-measures of the proposed stacked NSDAE for different states in Task 4.

Table 8

Hyper-parameters of the proposed stacked NSDAE in Task 4

Descriptions of hyper-parameters	Value
Dimension of the first hidden layer	450
Dimension of the second hidden layer	250
Dimension of the third hidden layer	100
Iteration number / Weight decay coefficient	60 / 0.004
Noise level/ Kernel width	0.08 / 1.2
Sparse constant / Sparse penalty factor	0.1 / 5
Initial learning rate / decay factor / momentum	0.01 / 1.1 / 0.8

5. Conclusions

To achieve accurate machine fault diagnosis with insufficient measured fault condition data, a digital twin-assisted deep transfer learning approach was presented in this paper. An NSDAE-based model was developed with improved activation and cost function. The simulation model of the physical asset was built and continuously updated when the system characteristics changed to form the DT model. Simulation data of all interested faults were generated from the DT to construct the training data in the

source domain. The NSDAE was first pre-trained using the data in the source domain, followed by one-shot learning with parameter transfer.

The fault diagnosis effectiveness of the proposed method was validated using simulation data of a triplex pump with various fault conditions. The results confirmed that DT and parameter transfer learning represent promising tools to overcome the difficulties in machine fault diagnosis when labeled fault condition samples were insufficient or unavailable. The experimental results showed the advantage of the stacked NSDAE over other state-of-art deep learning methods. Potential future work will focus on the combination strategy of DT and deep transfer learning, which can take full advantages of model mechanism and data information, including how to enhance diagnosis performance by fusing sensory data from multiple sources, how to improve the DT optimization algorithm to obtain a more precise model and how to achieve physics-informed machine learning for intelligent fault diagnosis.

Acknowledgments

This work was supported by the Horizon 2020 (No. 101007005), the National Natural Science Foundation of China (No. 51905160), the National Key Research and Development Program of China (No. 2020YFB1712103), the Natural Science Foundation of Hunan Province (No. 2020JJ5072) and the Fundamental Research Funds for the Central Universities (No. 531118010335).

Reference

[1] A. Kusiak, "Smart manufacturing," *Int. J. Prod. Res.*, vol. 56, no. 1–2, pp. 508–517, Jan. 2018, doi: 10.1080/00207543.2017.1351644.

[2] X. Li, J. Wan, H. Dai, M. Imran, M. Xia, and A. Celesti, "A Hybrid Computing Solution and Resource Scheduling Strategy for Edge Computing in Smart Manufacturing," *IEEE Trans. Ind. Informatics*, vol. 15, no. 7, pp. 4225–4234, 2019.

[3] W. Sun, R. Zhao, R. Yan, S. Shao, and X. Chen, "Convolutional Discriminative Feature Learning for Induction Motor Fault Diagnosis," *IEEE Trans. Ind. Informatics*, vol. 13, no. 3, pp. 1350–1359, 2017, doi: 10.1109/TII.2017.2672988.

[4] Y. Lei, B. Yang, X. Jiang, F. Jia, N. Li, and A. K. Nandi, "Applications of machine learning to machine fault diagnosis: A review and roadmap," *Mech. Syst. Signal Process.*, vol. 138, p. 106587, 2020, doi: https://doi.org/10.1016/j.ymssp.2019.106587.

[5] A. Glowacz, "Fault diagnosis of single-phase induction motor based on acoustic signals," *Mech. Syst. Signal Process.*, vol. 117, pp. 65–80, 2019, doi: https://doi.org/10.1016/j.ymssp.2018.07.044.

[6] S. Shao, S. McAleer, R. Yan, and P. Baldi, “Highly Accurate Machine Fault Diagnosis Using Deep Transfer Learning,” *IEEE Trans. Ind. Informatics*, vol. 15, no. 4, pp. 2446–2455, 2019, doi: 10.1109/TII.2018.2864759.

[7] C. Shen, Y. Qi, J. Wang, G. Cai, and Z. Zhu, “An automatic and robust features learning method for rotating machinery fault diagnosis based on contractive autoencoder,” *Eng. Appl. Artif. Intell.*, vol. 76, pp. 170–184, 2018, doi: https://doi.org/10.1016/j.engappai.2018.09.010.

[8] H. Shao, H. Jiang, H. Zhang, and T. Liang, “Electric Locomotive Bearing Fault Diagnosis Using a Novel Convolutional Deep Belief Network,” *IEEE Trans. Ind. Electron.*, vol. 65, no. 3, pp. 2727–2736, 2018, doi: 10.1109/TIE.2017.2745473.

[9] M. Xia, T. Li, L. Liu, L. Xu, and C. W. de Silva, “Intelligent fault diagnosis approach with unsupervised feature learning by stacked denoising autoencoder,” *IET Sci. Meas. Technol.*, vol. 11, no. 6, pp. 687–695, 2017, doi: 10.1049/iet-smt.2016.0423.

[10] Z. Chen, K. Gryllias, and W. Li, "Mechanical fault diagnosis using Convolutional Neural Networks and Extreme Learning Machine," *Mech. Syst. Signal Process.*, vol. 133, pp. 106272, 2019, doi: https://doi.org/10.1016/j.ymssp.2019.106272.

[11] H. Shao, H. Jiang, X. Li, and S. Wu, “Intelligent fault diagnosis of rolling bearing using deep wavelet auto-encoder with extreme learning machine,” *Knowledge-Based Syst.*, vol. 140, pp. 1–14, 2018, doi: https://doi.org/10.1016/j.knosys.2017.10.024.

[12] F. B. Abid, M. Sallem, and A. Braham, “Robust Interpretable Deep Learning for Intelligent Fault Diagnosis of Induction Motors,” *IEEE Trans. Instrum. Meas.*, vol. 69, no. 6, pp. 3506–3515, 2020, doi: 10.1109/TIM.2019.2932162.

[13] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, “A Survey on Deep Transfer Learning BT - Artificial Neural Networks and Machine Learning – ICANN 2018,” 2018, pp. 270–279.

[14] J. Li, R. Huang, G. He, Y. Liao, Z. Wang and W. Li, "A Two-Stage Transfer Adversarial Network for Intelligent Fault Diagnosis of Rotating Machinery with Multiple New Faults," IEEE/ASME Transactions on Mechatronics, 2020, doi: 10.1109/TMECH.2020.3025615.

[15] Z. He, H. Shao, X. Zhong and X. Zhao, "Ensemble transfer CNNs driven by multi-channel signals for fault diagnosis of rotating machinery cross working conditions," *Knowledge-Based Syst.*, vol. 207, pp. 106396, 2020, doi: 10.1016/j.knosys.2020.106396.

[16] L. Wen, X. Li, and L. Gao, “A transfer convolutional neural network for fault diagnosis based on ResNet-50,” *Neural Comput. Appl.*, vol. 32, no. 10, pp. 6111–6124, 2020, doi: 10.1007/s00521-019-04097-w.

[17] J. Zhu, N. Chen, and C. Shen, “A New Deep Transfer Learning Method for Bearing Fault Diagnosis Under Different Working Conditions,” *IEEE Sens. J.*, vol. 20, no. 15, pp. 8394–8402, 2020, doi: 10.1109/JSEN.2019.2936932.

[18] h. shao, M. Xia, G. Han, Y. Zhang, and J. Wan, “Intelligent fault diagnosis of rotor-bearing system under varying working conditions with modified transfer CNN and thermal images,” *IEEE Trans. Ind. Informatics*, p. 1, 2020, doi: 10.1109/TII.2020.3005965.

[19] R. Yan, F. Shen, C. Sun, and X. Chen, “Knowledge Transfer for Rotary Machine Fault Diagnosis,” *IEEE Sens. J.*, vol. 20, no. 15, pp. 8374–8393, 2020, doi: 10.1109/JSEN.2019.2949057.

[20] F. Tao *et al.*, “Digital twin-driven product design framework,” *Int. J. Prod. Res.*, vol. 57, no. 12, pp. 3935–3953, 2019, doi: 10.1080/00207543.2018.1443229.

[21] Y. Fang, C. Peng, P. Lou, Z. Zhou, J. Hu, and J. Yan, “Digital-Twin-Based Job Shop Scheduling Toward Smart Manufacturing,” *IEEE Trans. Ind. Informatics*, vol. 15, no. 12, pp. 6425–6435, 2019, doi: 10.1109/TII.2019.2938572.

[22] Y. Yi, Y. Yan, X. Liu, Z. Ni, J. Feng, and J. Liu, “Digital twin-based smart assembly process design and application framework for complex products and its case study,” *J. Manuf. Syst.*, 2020, doi: https://doi.org/10.1016/j.jmsy.2020.04.013.

[23] J. Wang, L. Ye, R. X. Gao, C. Li, and L. Zhang, “Digital Twin for rotating machinery fault diagnosis in smart manufacturing,” *Int. J. Prod. Res.*, vol. 57, no. 12, pp. 3920–3934, 2019, doi: 10.1080/00207543.2018.1552032.

[24] P. Jain, J. Poon, J. P. Singh, C. Spanos, S. R. Sanders, and S. K. Panda, “A digital twin approach for fault diagnosis in distributed photovoltaic systems,” *IEEE Trans. Power Electron.*, vol. 35, no. 1, pp. 940–956, 2020, doi: 10.1109/TPEL.2019.2911594.

[25] P. Ramachandran, B. Zoph, and Q. V. Le, “Searching for Activation Functions,” *6th Int. Conf. Learn. Represent. ICLR 2018 - Work. Track Proc.*, Oct. 2017, Accessed: Sep. 28, 2020. [Online]. Available: http://arxiv.org/abs/1710.05941.

[26] S. Miller, “Predictive Maintenance Using a Digital Twin.” https://uk.mathworks.com/company/newsletters/articles/predictive-maintenance-using-a-digital-twin.html (accessed Jun. 01, 2020).