Mechanical Systems and Signal Processing 115 (2019) 213–237

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

journal homepage: www.elsevier.com/locate/ymssp

Deep learning and its applications to machine health monitoring

Rui Zhao [a], Ruqiang Yan [a,*], Zhenghua Chen [b], Kezhi Mao [b], Peng Wang [c], Robert X. Gao [c]

[a] School of Mechanical Engineering, Xi'an Jiaotong University, China
[b] School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore
[c] Department of Mechanical and Aerospace Engineering, Case Western Reserve University, United States

ARTICLE INFO

Article history:
Received 3 February 2018
Received in revised form 18 April 2018
Accepted 27 May 2018
Available online 14 June 2018

Keywords:
Deep learning
Machine health monitoring
Big data

ABSTRACT

Since 2006, deep learning (DL) has become a rapidly growing research direction, redefining state-of-the-art performances in a wide range of areas such as object recognition, image segmentation, speech recognition and machine translation. In modern manufacturing systems, data-driven machine health monitoring is gaining in popularity due to the widespread deployment of low-cost sensors and their connection to the Internet. Meanwhile, deep learning provides useful tools for processing and analyzing these big machinery data. The main purpose of this paper is to review and summarize the emerging research work of deep learning on machine health monitoring. After the brief introduction of deep learning techniques, the applications of deep learning in machine health monitoring systems are reviewed mainly from the following aspects: Auto-encoder (AE) and its variants, Restricted Boltzmann Machines and its variants including Deep Belief Network (DBN) and Deep Boltzmann Machines (DBM), Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN). In addition, an experimental study on the performances of these approaches has been conducted, in which the data and code have been online. Finally, some new trends of DL-based machine health monitoring methods are discussed.

Contents

* Corresponding author.
E-mail address: rqyan@ieee.org (R. Yan).

https://doi.org/10.1016/j.ymssp.2018.05.050
0888-3270/

1. Introduction

Industrial Internet of Things (IoT) and data-driven techniques have been revolutionizing manufacturing by enabling computer networks to gather the huge amount of data from connected machines and turn the big machinery data into actionable information [1–3]. As a key component in modern manufacturing system, machine health monitoring has fully embraced the big data revolution. Compared to top-down modeling provided by the traditional physics-based models [4–6], data-driven machine health monitoring systems offer a new paradigm of bottom-up solution for detection of faults after the occurrence of certain failures (diagnosis) and predictions of the future working conditions and the remaining useful life (prognosis) [1,7]. As we all know, the complex and noisy working condition hinders the construction of physical models, which make the modelling of complex dynamic systems very difficult [8,9]. Most of these physics-based models are unable to be updated with on-line measured data, which limits their effectiveness and flexibility. On the other hand, with significant development of sensors, sensor networks and computing systems, data-driven machine health monitoring models have become more and more attractive. To extract useful knowledge and make appropriate decisions from big data, machine learning techniques have been regarded as a powerful solution. As the hottest subfield of machine learning, deep learning is able to act as a bridge connecting big machinery data and intelligent machine health monitoring.

As a branch of machine learning, deep learning attempts to model hierarchical representations behind data and classify (predict) patterns via stacking multiple layers of information processing modules in hierarchical architectures. Recently, deep learning has been successfully adopted in various areas such as computer vision, automatic speech recognition, natural language processing, audio recognition and bioinformatics [10–13]. In fact, deep learning is not a new idea, which even dates back to the 1940s [14,15]. The popularity of deep learning today can be contributed to the following aspects:

* *Increasing Computing Power*: the advent of graphics processor unit (GPU), the lowered cost of hardware, the better software infrastructure and the faster network connectivity all reduce the required running time of deep learning algorithms significantly. For example, as reported in [16], the time required to learn a four-layer Deep Belief Network with 100 million free parameters can be reduced from several weeks to around a single day.
* *Increasing Data Size*: there is no doubt that the era of Big Data is coming. Our activities are almost all digitized, recorded by computers and sensors, connected to Internet, and stored in cloud. As pointed out in [1] that in industry-related applications such as industrial informatics and electronics, almost 1000 exabytes are generated per year and a 20-fold increase can be expected in the next ten years. The study in [3] predicts that 30 billion devices will be connected by 2020. Therefore, the huge amount of data is able to offset the complexity increase behind deep learning and improve its generalization capability.
* *Advanced Deep Learning Research*: the first breakthrough of deep learning is the pre-training method in an unsupervised way [17], where Hinton proposed to pre-train one layer at a time via restricted Boltzmann machine (RBM) and then fine-tune using backpropagation in 2007. This has been proven to be effective to train multi-layer neural networks.

Considering the capability of deep learning to address large-scale data and learn multi-scale/multi-level/hierarchical representation, deep learning can be a powerful and effective solution for machine health monitoring systems (MHMS). Conventional data-driven MHMS usually consists of the following key parts: hand-crafted feature design, feature extraction/selection and model training. The right set of features are designed, and then provided to some shallow machine learning algorithms including Support Vector Machines (SVM), Naive Bayes (NB), logistic regression [18–20]. It is shown that the representation of the data, which is provided to the machine learning algorithms, limits the performance [21]. However, it is

difficult to design appropriate features and perform feature selection. To alleviate this issue, feature extraction/selection methods, which can be regarded as a kind of information fusion, are performed between hand-crafted feature design and classification/regression models [22–24]. However, manually designing features for a complex domain requires a great deal of human labor and cannot be updated on-line. Additionally, it also requires a significant amount of expertise from the practitioner, which is not always available. At last, the above three modules including feature design, feature extraction/selection and model training cannot be jointly optimized which may hinder the final performance of the whole system. Deep learning based MHMS (DL-based MHMS) aim to extract hierarchical representations from input data by building deep neural networks with multiple layers of non-linear transformations. Intuitively, one layer operation can be regarded as a transformation from input values to output values. Therefore, the application of one layer can learn a new representation of the input data and then, the stacking structure of multiple layers can enable MHMS to learn complex concepts out of simple concepts that can be constructed from raw input. In addition, DL-based MHMS achieve an end-to-end system, which can automatically learn internal representations from raw input and predict targets. Compared to conventional data driven MHMS, DL-based MHMS do not require extensive human labor and knowledge for hand-crafted feature design. All model parameters including feature module and pattern classification/regression module can be trained jointly. Therefore, DL-based models can be applied to addressing machine health monitoring in a very general way. For example, it is possible that the model trained for fault diagnosis problem can be used for prognosis by only replacing the top softmax layer with a linear regression layer requiring some re-training [25]. The comparison between conventional data-driven MHMS and DL-based MHMS is given in Table 1. A high-level illustration of the principles behind these three kinds of MHMS discussed above is shown in Fig. 1.

Deep learning models have several variants such as Auto-encoders [26], Deep Belief Network [27], Deep Boltzmann Machines [28], Convolutional Neural Networks [29] and Recurrent Neural Networks [30]. During recent years, various researchers have demonstrated success of these deep learning models in the application of machine health monitoring. This paper attempts to provide a wide overview on these latest DL-based MHMS works that impact the state-of-the-art technologies. Compared to these frontiers of deep learning including Computer Vision and Natural Language Processing, machine health monitoring community is catching up and has witnessed an emerging research. Therefore, the purpose of this study is to present researchers and engineers in the area of machine health monitoring system, a global view of this hot and active topic, and help them to acquire basic knowledge, quickly apply deep learning models and develop novel DL-based MHMS. The remainder of this paper is organized as follows. The basic information on these deep learning models mentioned above are given in Section 2. Then, Section 3 reviews applications of deep learning models on machine health monitoring. In Section 4, an experimental study has been conducted in a tool wear prediction task. Finally, Section 5 gives a brief summary of

Table 1
Summary on comparison between conventional data-driven MHMS and DL-based MHMS.

MHMS	
Conventional Data-driven Methods	Deep Learning Methods
Expert knowledge and extensive human labor required for Hand-crafted features *Individual modules are trained step-by-step* *Unable to model large-scale data*	*End-to-end structure without hand-crafted features* *All parameters are trained jointly* *Suitable for large-scale data*

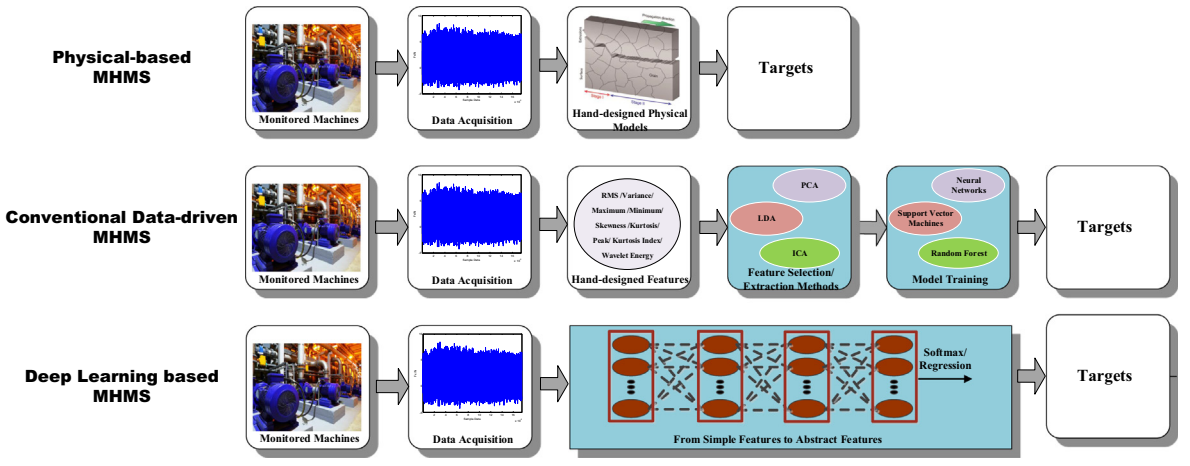

Fig. 1. Frameworks showing three different MHMS including Physical Models, Conventional Data-driven Models and Deep Learning Models. Shaded boxes denote data-driven components.

the recent achievements of DL-based MHMS and discusses some potential trends of deep learning in machine health monitoring.

As a survey paper, a comprehensive review of recent research efforts on deep learning based machine health monitoring system is given to present the whole picture of state-of-art related research for readers and foster the potential innovation in this field. In addition, implementations of several classical DL-based MHMS are public with which researchers/engineers can quickly understand and contribute to the field of DL-based MHMS research area.

2. Deep learning

Originated from artificial neural network, deep learning is a branch of machine learning which is featured by multiple non-linear processing layers and tries to learn hierarchical representations of data. Up to date, there are various deep learning architectures and this research topic is fast-growing, in which new models are being developed even every week. The community is quite open and there are a number of deep learning tutorials and books of good-quality [31,32]. Therefore, only a brief introduction to some major deep learning techniques that have been applied in machine health monitoring is given. In the following, four deep architectures including Auto-encoders, RBM, CNN, RNN and their corresponding variants are reviewed.

2.1. Auto-encoders (AE) and its variants

As a feed-forward neural network, auto-encoder consists of two phases including encoder and decoder, which is designed to learn a new representation of the data by trying to reconstruct the input data. Encoder takes an input $\mathbf{x}$ and transforms it to a hidden representation $\mathbf{h}$ via a non-linear mapping as follows:

$$\mathbf{h} = \varphi(\mathbf{W}\mathbf{x} + \mathbf{b}) \tag{1}$$

where φ is a non-linear activation function. The commonly used activation functions include *softmax*, *relu*, *tanh*, *sigmoid* and so on. Then, decoder maps the hidden representation back to the original representation in a similar way as follows:

$$\mathbf{z} = \varphi(\mathbf{W}'\mathbf{h} + \mathbf{b}') \tag{2}$$

Model parameters including $\theta = [\mathbf{W}, \mathbf{b}, \mathbf{W}', \mathbf{b}']$ are optimized to minimize the reconstruction error between $\mathbf{z} = f_\theta(\mathbf{x})$ and $\mathbf{x}$. One commonly adopted measure for the average reconstruction error over a collection of N data samples is squared error and the corresponding optimization problem can be written as follows:

$$\min_\theta \frac{1}{N}\sum_i^N \|\mathbf{x}_i - f_\theta(\mathbf{x}_i)\|_2^2 \tag{3}$$

where $\mathbf{x}_i$ is the i-th sample. It is clearly shown that AE can be trained in an unsupervised way. The hidden representation $\mathbf{h}$ can be regarded as a more abstract and meaningful representation for data sample $\mathbf{x}$.

2.1.1. Addition of sparsity

To prevent the learned transformation to be the identity one and regularize auto-encoders, the sparsity constraint is imposed on the hidden units [33]. The corresponding optimization function is updated as:

$$\min_\theta \frac{1}{N}\sum_i^N \|\mathbf{x}_i - f_\theta(\mathbf{x}_i)\|_2^2 + \beta\sum_j^m KL(p||p_j) \tag{4}$$

where m is the hidden layer size, the second term is the summation of the KL-divergence over the hidden units and β is a controlling weight for the sparsity penalty term. The KL-divergence on j-th hidden neuron is given as:

$$KL(p||p_j) = p\log\left(\frac{p}{p_j}\right) + (1-p)\log\left(\frac{1-p}{1-p_j}\right) \tag{5}$$

where p is the predefined mean activation target and p_j is the average activation of the j-th hidden neuron over the entire dataset. Given a small p, the addition of sparsity constraint can lead the learned hidden representation to be a sparse representation. Therefore, the variant of AE is named as sparse auto-encoder.

2.1.2. Addition of denoising

Different from conventional AE, denoising AE takes a corrupted version of data as input and is trained to reconstruct/denoise the clean input $\mathbf{x}$ from its corrupted sample $\tilde{\mathbf{x}}$. The most commonly adopted noise is dropout noise/binary masking noise, which randomly sets a fraction of the input features to be zero [26]. The variant of AE is denoising auto-encoder (DA), which can learn more robust representation and prevent it from learning the identity transformation.

2.1.3. Stacking structure

Several DA can be stacked together to form a deep network and learn representations by feeding the outputs of the l-th layer as inputs to the $(l+1)$-th layer [26]. And the training is done one layer greedily at a time.

Since auto-encoder can be trained in an unsupervised way, auto-encoder, especially stacked denoising auto-encoder (SDA), can provide an effective pre-training solution via initializing the weights of deep neural network (DNN) to train the model. After layer-wise pre-training of SDA, the parameters of auto-encoders can be set to the initialization for all the hidden layers of DNN. Then, the supervised fine-tuning is performed to minimize prediction error on a labeled training data. Usually, a softmax/regression layer is added on top of the network to map the output of the last layer in AE to targets. The whole process is shown in Fig. 2. The pre-training protocol based on SDA can make DNN models have better convergence capability compared to arbitrary random initialization. It should be noted that training deep neural networks often suffers from gradient vanishing/exploding problems due to these commonly adopted *tanh* or *sigmoid* nonlinear activation functions. Therefore, unsupervised training enabled by AE is meaningful and powerful. However, *relu* activation relieved this problem, which was proposed in 2012. Supervised training of deep neural networks such as deep convolutional neural network and recurrent neural network became possible (see Fig. 3).

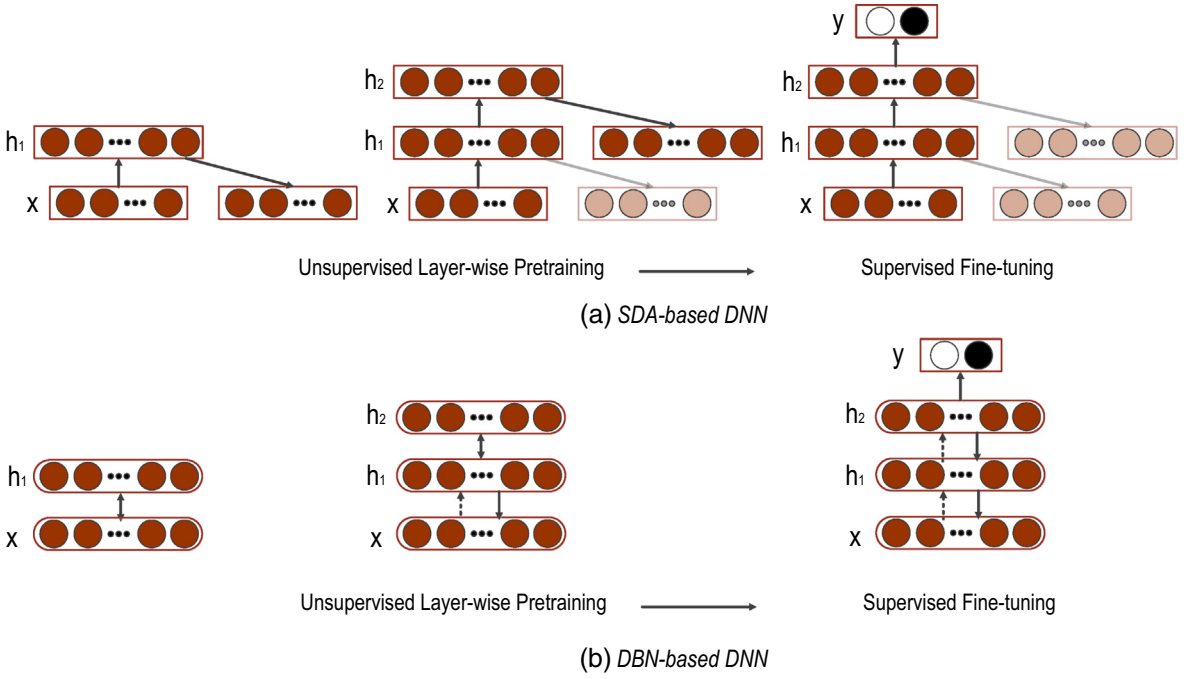

Fig. 2. Illustrations for unsupervised pre-training and supervised fine-tuning of SAE-DNN (a) and DBN-DNN (b).

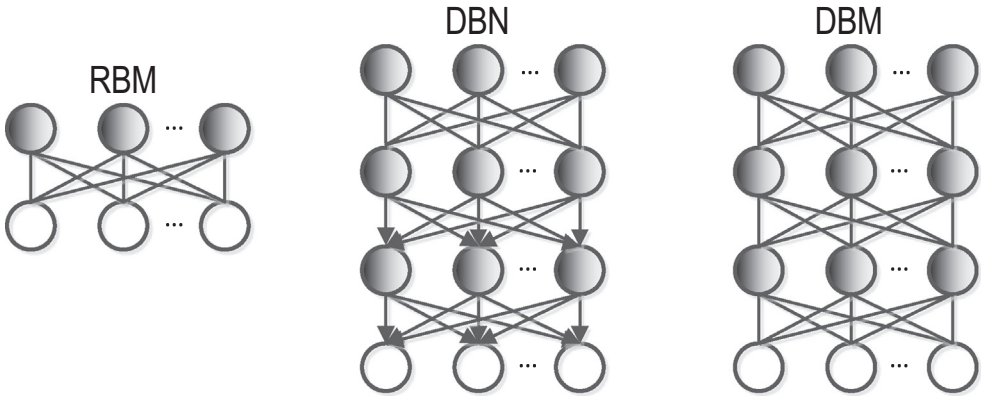

Fig. 3. Frameworks showing RBM, DBN and DBM. Shaded boxes denote hidden units.

2.2. RBM and its variants

As a special type of Markov random field, restricted Boltzmann machine (RBM) is a two-layer neural network forming a bipartite graph that consists of two groups of units including visible units $\mathbf{v}$ and hidden units $\mathbf{h}$ under the constraint that there exists a symmetric connection between visible units and hidden units and there are no connections between nodes with a group.

Given the model parameters $\theta = [\mathbf{W}, \mathbf{b}, \mathbf{a}]$, the energy function can be given as:

$$E(\mathbf{v}, \mathbf{h}; \theta) = -\sum_{i=1}^{I}\sum_{j=1}^{J} w_{ij} v_i h_j - \sum_{i=1}^{I} b_i v_i - \sum_{j=1}^{J} a_j h_j \tag{6}$$

that w_{ij} is the connecting weight between visible unit v_i, whose total number is I and hidden unit h_j whose total number is J, b_i and a_j denote the bias terms for visible units and hidden units, respectively. The joint distribution over all the units is calculated based on the energy function $E(\mathbf{v}, \mathbf{h}; \theta)$ as:

$$p(\mathbf{v}, \mathbf{h}; \theta) = \frac{exp(-E(\mathbf{v}, \mathbf{h}; \theta))}{Z} \tag{7}$$

where $Z = \sum_{\mathbf{h},\mathbf{v}} exp(-E(\mathbf{v}, \mathbf{h}; \theta))$ is the partition function or normalization factor. Then, the conditional probabilities of hidden and visible units $\mathbf{h}$ and $\mathbf{v}$ can be calculated as:

$$p(h_j = 1 | v; \theta) = \delta\left(\sum_{i=1}^{I} w_{ij} v_i + a_j\right) \tag{8}$$

$$p(v_i = 1 | v; \theta) = \delta\left(\sum_{j=1}^{J} w_{ij} h_j + b_i\right) \tag{9}$$

where δ is defined as a logistic function, i.e., $\delta(x) = \frac{1}{1+exp(x)}$. RBM is trained to maximize the joint probability. The learning of $\mathbf{W}$ is done through a method called contrastive divergence (CD) [34].

2.2.1. Deep belief network

Deep belief network (DBN) can be constructed by stacking multiple RBMs, where the output of the l-th layer (hidden units) is used as the input of the $(l+1)$-th layer (visible units) [35]. Similar to SDA, DBN can be trained in a greedy layer-wise unsupervised way. After pre-training, the parameters of this deep architecture can be further fine-tuned with respect to a proxy for the DBN log-likelihood, or with respect to labels of training data by adding a softmax layer as the top layer, which is shown in Fig. 2(b).

2.2.2. Deep Boltzmann Machine

Deep Boltzmann machine (DBM) can be regarded as a deep structured RMBs where hidden units are grouped into a hierarchy of layers instead of a single layer [28]. Following the RMB's connectivity constraint, there is only full connectivity between subsequent layers and no connections within layers or between non-neighbouring layers are allowed. The main difference between DBN and DBM lies that DBM is fully undirected graphical model, while DBN is mixed directed/undirected one. Different from DBN that can be trained layer-wisely, DBM is trained as a joint model. Therefore, the training of DBM is more computationally expensive than that of DBN.

2.3. Convolutional neural network

Convolutional neural networks (CNNs) were firstly proposed by LeCun [36] for image processing, which is featured by two key properties: spatially shared weights and spatial pooling. CNN models have shown their success in various computer vision applications [36–38] where input data are usually 2D data. CNN has also been introduced to address sequential data including Natural Language Processing and Speech Recognition [39,40].

CNN aims to learn abstract features by alternating and stacking convolutional layers and pooling layers. In CNN, the convolutional layers (convolutional kernels) convolve multiple local filters with raw input data and generate translation-invariant local features and the subsequent pooling layers extract features with a fixed-length over sliding windows of the raw input data following several rules such as average, max and so on. Considering that 2D-CNN has been illustrated extensively in previous research compared to 1D-CNN, here, only the mathematical details behind 1D-CNN is given as follows:

Firstly, we assume that the input sequential data is $\mathbf{x} = [\mathbf{x}_1, \ldots, \mathbf{x}_T]$ that T is the length of the sequence and $\mathbf{x}_i \in \mathbb{R}^d$ at each time step.

2.3.1. Convolution

the dot product between a filter vector $\mathbf{u} \in \mathbb{R}^{md}$ and an concatenation vector representation $\mathbf{x}_{i:i+m-1}$ defines the convolution operation as follows:

$$c_i = \varphi(\mathbf{u} \cdot \mathbf{x}_{i:i+m-1} + b) \tag{10}$$

where $\cdot$ represents the dot product, b and φ denote bias term and non-linear activation function, respectively. $\mathbf{x}_{i:i+m-1}$ is a m-length window starting from the i-th time step, which is described as:

$$\mathbf{x}_{i:i+m-1} = \mathbf{x}_i \oplus \mathbf{x}_{i+1} \oplus \ldots \oplus \mathbf{x}_{i+m-1} \tag{11}$$

where $\oplus$ is the concatenation operation of two vectors. As defined in Eq. (10), the output scale c_i can be regarded as the activation of the filter $\mathbf{u}$ on the corresponding subsequence $\mathbf{x}_{i:i+m-1}$. By sliding the filtering window from the beginning time step to the ending time step, a feature map as a vector can be given as follows:

$$\mathbf{c}_j = [c_1, c_2, \ldots, c_{l-m+1}] \tag{12}$$

where the index j represents the j-th filter. It corresponds to multi-windows as $\{\mathbf{x}_{1:m}, \ldots, \mathbf{x}_{l-m+1:l}\}$.

2.3.2. Pooling

Pooling layer is able to reduce the length of the feature map, which can further minimize the number of model parameters. These commonly adopted pooling operations include max and average pooling. In the following, max pooling is explained in details. The hyper-parameter of pooling layer is pooling length denoted as s. The max operation is to take a max over the s consecutive values in feature map $\mathbf{c}_j$.

Then, the compressed feature vector can be obtained as:

$$\mathbf{h} = \left[h_1, h_2, \ldots, h_{\frac{l-m}{s}+1}\right] \tag{13}$$

where $h_j = \max(c_{(j-1)s}, c_{(j-1)s+1}, \ldots, c_{js-1})$. Then, via alternating the above two layers: convolution and max-pooling ones, fully connected layers and a softmax layer are usually added as the top layers to make predictions. To give a clear illustration, the framework for a one-layer CNN has been displayed in Fig. 4.

2.4. Recurrent neural network

As stated in [14], recurrent neural networks (RNNs) are the deepest of all neural networks, which can generate and address memories of arbitrary-length sequences of input patterns. RNN is able to build connections between units from a

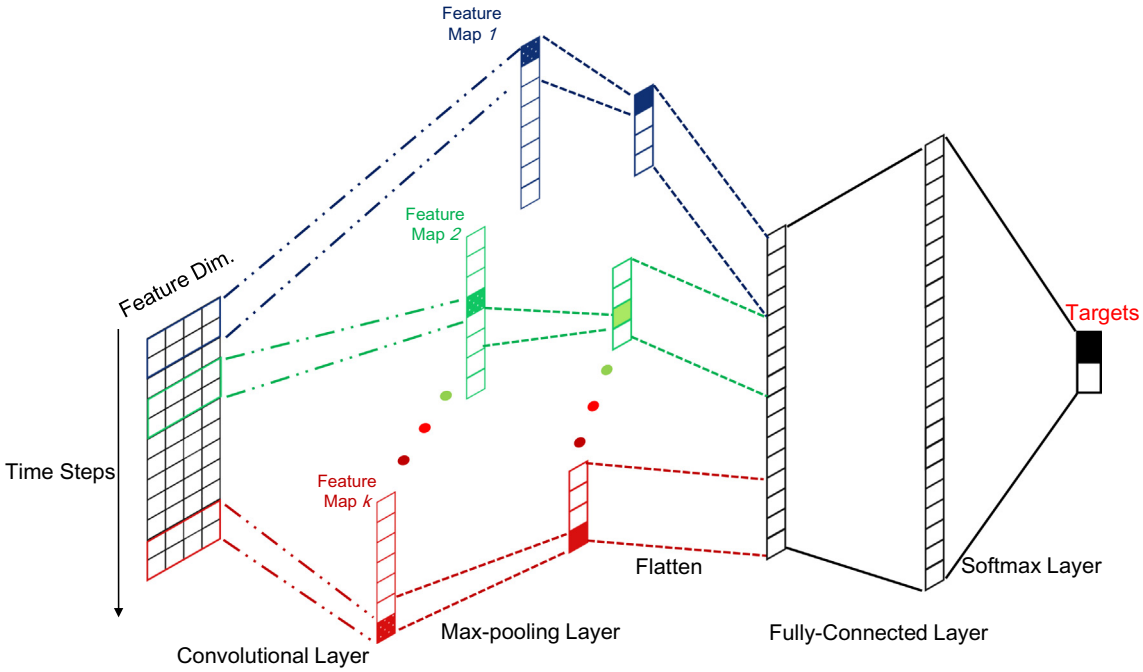

Fig. 4. Illustrations for one-layer CNN that contains one convolutional layer, one pooling layer, one fully-connected layer, and one softmax layer.

directed cycle. Different from basic neural network: multi-layer perceptron that can only map from input data to target vectors, RNN is able to map from the entire history of previous inputs to target vectors in principle and allow a memory of previous inputs to be kept in the network's internal state. RNNs can be trained via backpropagation through time for supervised tasks with sequential input data and target outputs [41,30,42].

RNN can address the sequential data using its internal memory, as shown in Fig. 5(a). The transition function defined in each time step t takes the current time information $\mathbf{x}_t$ and the previous hidden output $\mathbf{h}_{t-1}$ and updates the current hidden output as follows:

$$\mathbf{h}_t = \mathbb{H}(\mathbf{x}_t, \mathbf{h}_{t-1}) \tag{14}$$

where $\mathbb{H}$ defines a nonlinear and differentiable transformation function. After processing the whole sequence, the hidden output at the last time step, i.e., $\mathbf{h}_T$, is the learned representation of the input sequential data whose length is T. A conventional Multilayer perceptron (MLP) is added on top to map the obtained representation $\mathbf{h}_T$ to targets.

Various transition functions can lead to various RNN models. The most simple one is vanilla RNN that is given as follows:

$$\mathbf{h}_t = \varphi(\mathbf{W}\mathbf{x}_t + \mathbf{H}\mathbf{h}_{t-1} + \mathbf{b}) \tag{15}$$

where $\mathbf{W}$ and $\mathbf{H}$ denote transformation matrices and $\mathbf{b}$ is the bias vector. And φ denotes the nonlinear activation function such as *sigmoid* and *tanh* functions. Due to the vanishing gradient problem during backpropagation for model training, vanilla RNN may not capture long-term dependencies. Therefore, Long-short term memory (LSTM) and gated recurrent units (GRU) were presented to prevent backpropagated errors from vanishing or exploding [43–47]. The core idea behind these advanced RNN variants is that gates are introduced to avoid the long-term dependency problem and enable each recurrent unit to adaptively capture dependencies of different time scales.

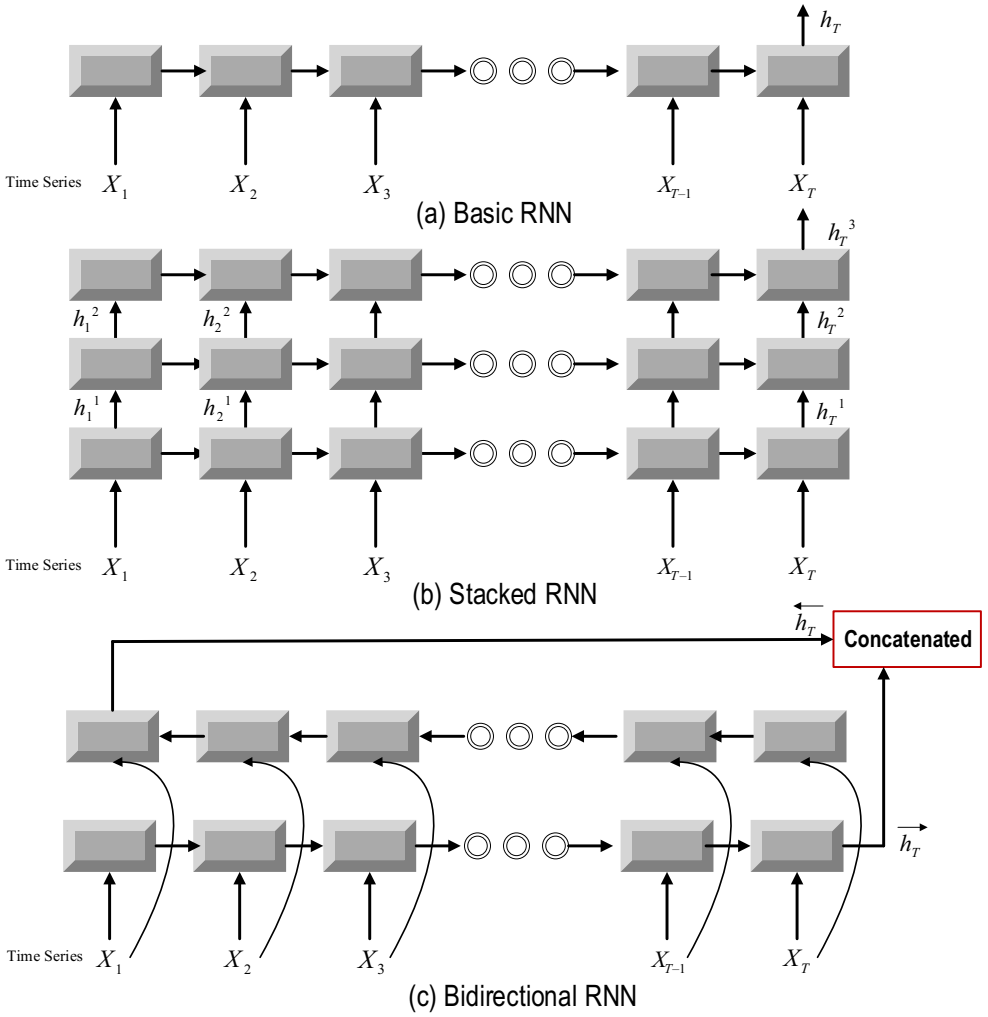

Fig. 5. Illustrations of normal RNN, stacked RNN and bidirectional RNN.

Besides these proposed advanced transition functions such as LSTMs and GRUs, multi-layer and bi-directional recurrent structure can increase the model capacity and flexibility. As shown in Fig. 5(b), multi-layer structure can enable the hidden output of one recurrent layer to be propagated through time and used as the input data to the next recurrent layer. The bidirectional recurrent structure is able to process the sequence data in two directions including forward and backward ways with two separate hidden layers, which is illustrated in Fig. 5(c). The following equations define the corresponding hidden layer function and the $\rightarrow$ and $\leftarrow$ denote forward and backward processes, respectively.

$$\begin{aligned}\overrightarrow{\mathbf{h}}_t &= \overrightarrow{\mathbb{H}}(\mathbf{x}_t, \overrightarrow{\mathbf{h}}_{t-1}),\\ \overleftarrow{\mathbf{h}}_t &= \overleftarrow{\mathbb{H}}(\mathbf{x}_t, \overleftarrow{\mathbf{h}}_{t+1}).\end{aligned} \tag{16}$$

Then, the final vector $\mathbf{h}_T$ is the concatenated vector of the outputs of forward and backward processes as follows:

$$\mathbf{h}_T = \overrightarrow{\mathbf{h}}_T \oplus \overleftarrow{\mathbf{h}}_1 \tag{17}$$

2.5. Optimization methods for neural networks

To learn optimal parameters for neural network models, gradient descent is the most widely used method. Gradient descent is performed to minimize an objective function $J(\theta)$ by updating the parameters θ in the opposite direction of the gradient of the object function $\nabla J(\theta)$ w.r.t to the parameters, in which the learning rate is used to control the size of the steps to reach a minimum. An overview of various gradient descent methods optimization for deep learning models is given here [48]. However, one tricky point lies that the weight initialization will influence the convergence so that it is necessary to select appropriate model parameters initialization scheme [49].

3. Applications of deep learning in machine health monitoring

The conventional MLP has been applied in the field of machine health monitoring for many years [50–53]. The deep learning techniques have recently been applied to a large number of machine health monitoring systems. The layer-by-layer pretraining of DNN based on AE or RBM can facilitate the training of DNN and improve its discriminative power to characterize machinery data. CNN and RNN provide more advanced and complex composition mechanism to learn representations from machinery data. In these DL-based MHMS systems, the top layer normally represents the targets. For diagnosis where targets are discrete values, softmax layer is applied. For prognosis with continuous targets, liner regression layer is added. What is more, the end-to-end structure enables DL-based MHMS to be constructed with less human labor and expert knowledge, therefore these models are not limited to specific equipment or domain. In the following, a brief survey of DL-based MHMS is presented in these above four DL architectures: AE, RBM, CNN and RNN.

3.1. AE and its variants for machine health monitoring

AE models, especially stacked DA, can learn representations from machinery data in an automatic way. Sun et al. proposed a one layer AE-based neural network to classify induction motor faults [54]. Due to the limited size of training data, they focused on preventing overfitting. Not only the number of hidden layer was set to 1, but also dropout technique that masks portions of output neurons randomly was applied on the hidden layer. The whole model has been shown in Fig. 6. The majority of proposed models are based on deep architectures by stacking multiple auto-encoders. For example, Lu et al. presented a detailed empirical study of stacked denoising autoencoders with three hidden layers for fault diagnosis of rotating machinery components [55]. Specifically, in their experiments including single working condition that training and testing data share one operation condition and cross working conditions that training and testing data are sampled from two different operation conditions, the effectiveness of deep architecture, sparsity constraint and denoising operation in the SDA model were evaluated. They recommended that three hidden layers with sparsity criterion of 0.15 and destruction level of 0.25 is optimal. In [56], different structures of a two-layer SAE-based DNN were designed by varying hidden layer size and its masking probability, and evaluated for their performances in fault diagnosis.

In these above works, the input features to AE models are raw sensory time-series. Therefore, the input dimensionality is always over hundred, even one thousand. The possible high dimensionality may lead to some potential concerns such as heavy computation cost and overfiting caused by huge model parameters. Therefore, some researchers focused on AE models built upon features extracted from raw input. Jia et al. fed the frequency spectra of time-series data into SAE for rotating machinery diagnosis [57], considering the frequency spectra is able to demonstrate how their constitutive components are distributed with discrete frequencies and may be more discriminative over the health conditions of rotating machinery. In [58], Sun et al. utilized compressed sensing techniques to extract low-dimensional features from raw time-series signal as input features into SAE-DNN models. In [59], Zhou et al. proposed three cascaded SAE-DNNs that each module is for mode partition classification, fault source location classification and fault severity recognition, respectively. The input features are frequency coefficients based on Fast Fourier Transform. Tan et al. used digital wavelet frame and nonlinear soft threshold method to process the vibration signal and built a SAE on the preprocessed signal for roller bearing fault diagnosis [60].

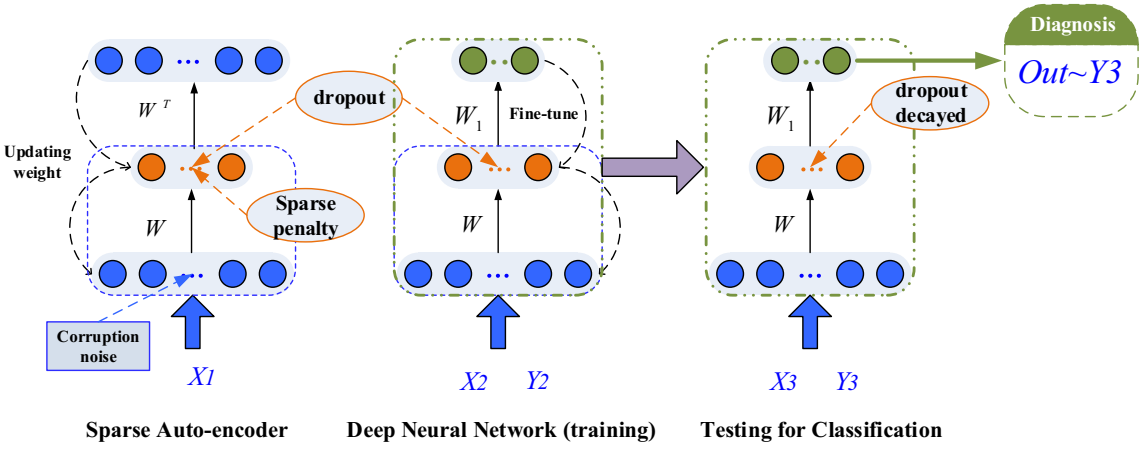

Fig. 6. Illustrations of the proposed sparse autoencoder for induction motor fault diagnosis in [54].

Zhu et al. proposed a SAE-based DNN for hydraulic pump fault diagnosis with input as frequency domain features after Fourier transform [61]. In experiments, *relu* activation and dropout technique were analyzed and experimental results have shown to be effective in preventing gradient vanishing and overfiting. In the work presented in [62], the normalized spectrogram generated by STFT of sound signal was fed into two-layers SAE-based DNN for rolling bearing fault diagnosis. Galloway et al. built a two layer SAE-based DNN on spectrograms generated from raw vibration data for tidal turbine vibration fault diagnosis [63]. A SAE-based DNN with input as principal components of data extracted by principal component analysis was proposed for spacecraft fault diagnosis in [64]. Multi-domain statistical features including time domain features, frequency domain features and time–frequency domain features were fed into the SAE framework, which can be regarded as one kind of feature fusion [65]. Similarly, Verma et al. also used these three domains features to fed into a SAE-based DNN for fault diagnosis of air compressors [66] and Sohaib et al. proposed SAE-DNN model based on these three domains features for bearing fault diagnosis[67]. In [68], Chen et al. fed tri-domain features into SAE and adopted support vector machine as the final classifier. To overcome the issue of overfitting problem, Chen et al. adopted data augmentation technique by adding Gaussian noise into training data [69].

Except for these applied multi-domain features, multi-sensory data are also addressed by SAE models. Reddy et al. utilized SAE to learn representation on raw time series data from multiple sensors for anomaly detection and fault disambiguation in flight data. To address multi-sensory data, synchronized windows were firstly traversed over multi-modal time series with overlap, and then windows from each sensor were concatenated as the input to the following SAE [70]. In [71], SAE was leveraged for multi-sensory data fusion and the followed DBN was adopted for bearing fault diagnosis, which achieved promising results. The statistical features in time domain and frequency domain extracted from the vibration signals of different sensors were adopted as inputs to a two-layer SAE with sparsity constraint neural networks. The learned representations were fed into a deep belief network for pattern classification.

In addition, some variants of the conventional SAE were proposed or introduced for machine health monitoring. In [72], Thirukovalluru et al. proposed a two-phase framework that SAE only learns representation and other standard classifiers such as SVM and random forest perform classification. Specifically, in SAE module, handcrafted features based on FFT and wavelet packet transform (WPT) were fed into SAE-based DNN. After pre-training and supervised fine-tuning which includes two separated procedures: softmax-based and Median-based fine-tuning methods, the extensive experiments on five datasets including air compressor monitoring, drill bit monitoring, bearing fault monitoring and steel plate monitoring demonstrated the generalization capability of DL-based machine health monitoring systems. Wang et al. proposed a novel continuous sparse auto-encoder (CSAE) as an unsupervised feature learning for transformer fault recognition [73]. Different from conventional sparse AE, their proposed CSAE added the stochastic unit into activation function of each visible unit as:

$$s_j = \varphi_j\left(\sum_i w_{ij}x_i + a_i + \sigma N_j(0,1)\right) \tag{18}$$

where s_j is the output corresponding to the input x_i, w_{ij} and a_i denote model parameters, φ_j represents the activation function and the last term $\sigma N_j(0,1))$ is the added stochastic unit, which is a zero-mean Gaussian with variance σ^2. The incorporation of stochastic unit is able to change the gradient direction and prevent overfitting. Mao et al. adopted a variant of AE named Extreme Learning Machine-based auto-encoder for bearing fault diagnosis, which is more efficient than conventional SAE models without sacrificing accuracies in fault diagnosis [74]. Different from AE that was trained via back-propagation, the transformation in encoder phase was randomly generated and the one in decoder phase was learned in a single step via least-squares fit [75]. In [76], Jia et al. pointed out that two potential shortcomings behind traditional autoencoders such as similar features learning and shift variant properties hinder the performance on automatic feature extraction of mechanical signals. Therefore, they proposed normalized sparse autoencoder (NSAE) by adding rectified liner units as activation func-

tion, abandoning bias and adopting L2 norm instead of KL divergence function in formulation of autoencoder. Then, a Local Connection Network (LCN) which takes the average value of NSAE's outputs was adopted to extract shift-invariant features. To match the features of the complex signal, Shao et.al replaced the original MSE loss function with the maximum correlated entropy in their designed autoencoders and adopted artificial fish swarm algorithm to select its key parameters [77]. In their another work, an ensemble model of autoencoders with 15 different activation functions was proposed in [78], in which the ensemble scheme is based on majority voting with different weights. They also proposed a fusion scheme of two autoencoders including denoising auto-encoder (DAE) and contractive auto-encoder (CAE) based on locality preserving projection (LPP) in [79]. Li et al. proposed a fully-connected winner-take-all autoencoder [80] (FCWTA) for bearing fault diagnosis. Their model's novelty lies in two aspects: one is about lifetime virtualization and deep recognition for system fault classification sparsity that only k largest activations of each hidden nodes among all samples in a mini-batch are kept and the other is the ensemble framework that the input signal is segmented into several windows and each window is fed into FCWTA to obtain prediction results. Finally, a soft voting method was implemented to fuse all prediction results.

In addition, Lu et al. focused on the visualization of learned representation by a two-layer SAE-based DNN, which provides a novel view to evaluate the DL-based MHMS [81]. In their paper, the discriminative power of learned representation can be improved with the increasing of layers.

3.2. RBM and its variants for machine health monitoring

In the section, some works focus on developing RBM to learn representation from machinery data. Most of works introduced here are based on deep belief networks (DBN) that can pretrain a deep neural network (DNN).

In [82], a RBM based method for bearing remaining useful life (RUL) prediction was proposed. Linear regression layer was added at the top of RBM after pretraining to predict the future root mean square (RMS) based on a lagged time series of RMS values. Then, RUL was calculated by using the predicted RMS and the total time of the bearing's life. In their another work [83], they utilized similar structure: DBN-FNN to predict RUL value directly. Liao et al. proposed a new RBM for representation learning to predict RUL of machines [84]. In their work, a new regularization term modeling the trendability of the hidden nodes was added into the training objective function of RBM. Then, unsupervised self-organizing map algorithm (SOM) was applied to transforming the representation learned by the enhanced RBM to one scale named health value. Finally, the health value was used to predict RUL via a similarity-based life prediction algorithm. In [85], a multi-modal deep support vector classification approach was proposed for fault diagnosis of gearboxes. Firstly, three modalities features including time, frequency and time–frequency ones were extracted from vibration signals. Then, three Gaussian-Bernoulli deep Boltzmann machines (GDBMS) were applied to addressing the above three modalities, respectively. In each GDBMS, the softmax layer was used at the top. After the pretraining and the fine-tuning processes, the probabilistic outputs of the softmax layers from these three GDBMS were fused by a support vector classification (SVC) framework to make final prediction. Li et al. applied one GDBMS directly on the concatenation feature consisting of three modalities features including time, frequency and time–frequency ones and stacked one softmax layer on top of GDBMS to recognize fault categories [86]. Li et al. adopted a two-layers DBM to learn deep representations of the statistical parameters of the WPT of raw sensory signal for gearbox fault diagnosis [87]. In this work focusing on data fusion, two DBMs were applied on acoustic and vibratory signals and random forest was applied to fuse the representations learned by these two DBMs. Shao et al. stacked multiple RBM into DBM model for fault diagnosis, whose input is frequency domain data based on Fast Fourier Transform (FFT) [88]. In [89], Zhang et al. utilized deep belief network for ball screw degradation recognition. The input features into DBN model are the fused frequency spectrum of various time domain signals in different sensors. In [90], Wang et al. proposed to use sliding-window spectrum feature (SWSF) as the input feature into DBN model for hydraulic fault diagnosis. In [91], time-domain and frequency-domain statistical features were extracted and fed into DBN. Then, the PSO-SVM was applied on DBN outputs for fault diagnosis. In [92], Wang et al. used two RBMs to form a DBM model to predict the material removal rate in polishing. Particle Swarm Optimization (PSO) algorithms were introduced to select hyperparameters such as DBN structure and learning rate. In [93], Chen et al. investigated the performances of several DNN-based models including DBM, DBN and SAE on four different preprocessing methods such as raw time domain signal, time domain feature, frequency domain feature and time–frequency domain feature. It is shown that these three DNN models are reliable and effective in fault diagnosis and the raw data-based DNN models perform worse compared to other three preprocessing methods. In [94], Gao et al. combined deep belief network and quantum inspired neural network (QINN) for aircraft fuel system fault diagnosis. The input features into DBN consist of time-domain feature and frequency-domain feature. The outputs of DBN were fed into quantum inspired neural network (QINN), which applies linear superposition of multiple DBNs with quantum intervals. In [95], Oh et al. applied DBN on vibration images to extract features and conducted final classification. The vibration image were generated from vibration sensor signal and Histogram of Oriented Gradients (HOG) was applied on the vibration image as input features into the following DBN.

Making use of DBN-based DNN, Ma et al. presented this framework for degradation assessment under a bearing accelerated life test [96]. The statistical feature, root mean square (RMS) fitted by Weibull distribution that can avoid areas of fluctuation of the statistical parameter and the frequency domain features were extracted as raw input. Beside the evaluation of the final classification accuracies, t-SNE algorithm was adopted to visualize the learned representation of DBN and outputs of each layer in DBN. They found the addition of hidden layer can increase the discriminative power in the learned representation. Shao et al. proposed DBN for induction motor fault diagnosis in [97]. As shown in Fig. 7, fast fourier transform was applied on raw time series data and the frequency-domain feature were fed into DBN models. Fu et al. employed deep belief

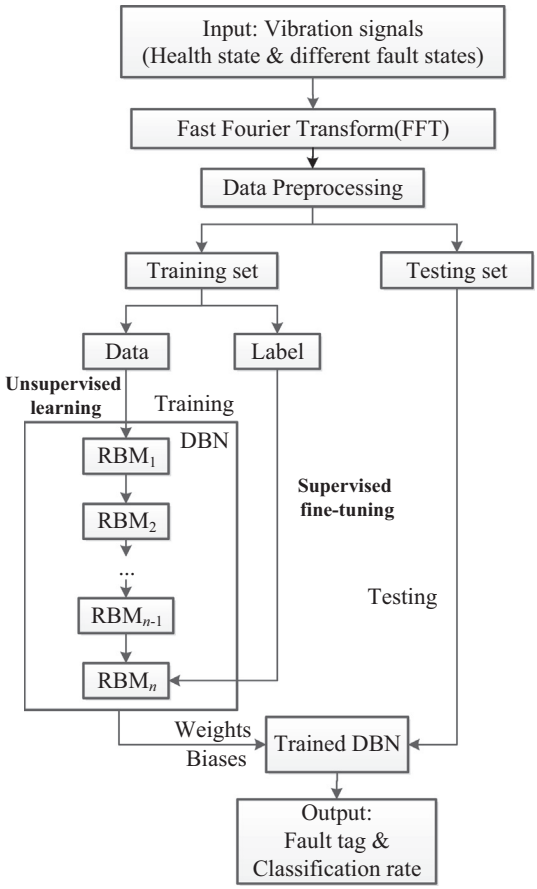

Fig. 7. Illustrations of DBN for induction motors fault diagnosis in [97].

networks for cutting states monitoring [98]. In the presented work, three different feature sets including raw vibration signal, Mel-frequency cepstrum coefficient (MFCC) and wavelet features were fed into DBN as three corresponding different inputs, which were able to achieve robust comparative performance on the raw vibration signal without too much feature engineering. Tamilselvan et al. proposed a multi-sensory DBN-based health state classification model. The model was verified in benchmark classification problems and two health diagnosis applications including aircraft engine health diagnosis and electric power transformer health diagnosis [99,100]. Tao et al. proposed a DBN based multisensor information fusion scheme for bearing fault diagnosis [101]. Firstly, 14 time-domain statistical features extracted from three vibration signals acquired by three sensors were concatenated together as an input vector to the DBM model. During pre-training, a predefined threshold value was introduced to determine its iteration number. In [102], a feature vector consisting of load and speed measure, time domain features and frequency domain features was fed into DBN-based DNN for gearbox fault diagnosis. In the work of [103], Gan et al. built a hierarchical diagnosis network for fault pattern recognition of rolling element bearings consisting of two consecutive phases where the four different fault locations (including one health state) were firstly identified and then discrete fault severities in each fault condition were classified. In each phase, the frequency-band energy features generated by WPT were fed into DBN-based DNN for pattern classification. In [104], raw vibration signals were pre-processed to generate 2D image based on omnidirectional regeneration (ODR) techniques and then, histogram of original gradients (HOG) descriptor was applied on the generated image and the learned vector was fed into DBN for automatic diagnosis of journal bearing rotor systems. Zhang et al. proposed an ensemble of DBNs with multi-objective evolutionary optimization on decomposition algorithm (MOEA/D) for fault diagnosis with multivariate sensory data [105]. DBNs with different architectures can be regarded as base classifiers and MOEA/D was introduced to adjust the ensemble weights to achieve a trade-off between accuracy and diversity. Zhang et al. then extended this above framework for one specific prognostics task: the RUL estimation of the mechanical system [106].

3.3. CNN for machine health monitoring

In some scenarios, machinery data can be presented in a 2D format such as time–frequency spectrum, while in some scenarios, they are in a 1D format, i.e., time-series. Therefore, CNN models are able to learn complex and robust representation

via its convolutional layer. Intuitively, filters in convolutional layers can extract local patterns in raw data and stacking these convolutional layers can further build complex patterns. Liu et al. proposed a dislocated time series convolutional neural network (DTS-CNN) for fault diagnosis of electric machine [107]. In their work, a dislocated time series layer was introduced to dislocate the 1D input mechanical signal into an output matrix. The DTS operation is intuitive that several intercepted signals from the original signal were arranged to form a matrix. Then, a conventional deep CNN model was applied. In [108], 2D CNN was introduced for gearbox fault diagnosis. As shown in Fig. 8, wavelet anaylsis was condcuted to transfer raw sensory input into 2D time–frequency images and a deep convolutional neural network was adopted for gearbox fault diagnosis. Janssens et al. utilized a 2D-CNN model for four categories rotating machinery conditions recognition, whose input was DFT results of two accelerometer signals from two sensors that were placed perpendicular to each other. Therefore, the height of input is the number of sensors. The adopted CNN model consisited of one convolutional layer and one fully connected layer. Then, the top softmax layer was adopted for classification [109]. Lu et al. rearranged the raw time series data into a 2D map based on slipping insertion whose size is 20 by 20 [110]. In [111], Babu et al. built a 2D deep convolution neural network to predict the RUL of system based on normalized-variate time series from sensor signals, in which one dimension of the 2D input is number of sensors as the setting reported in [109]. In their model, average pooling was adopted instead of max pooling. Since RUL is a continuous value, linear regression was used on the top layer. Ding et al. proposed a deep Convolutional Network (ConvNet) where wavelet packet energy (WPE) image was used as input for spindle bearing fault diagnosis [112]. To fully discover the hierarchical representation, a multiscale layer was added after the last convolutional layer, which concatenates the outputs of the last convolutional layer and the ones of the previous pooling layer. Guo et al. proposed a hierarchical adaptive deep convolution neural network (ADCNN) [113]. Firstly, the input time series data as a signal-vector was transformed into a 32×32 matrix, which follows the typical input format adopted by LeNet [114]. In addition, they designed a hierarchical framework to recognize fault patterns and fault size. In the fault pattern decision module, the first ADCNN was adopted to recognize fault type. In the fault size evaluation layer, based on each fault type, ADCNN with the same structure was used to predict fault size. Here, the classification mechanism is still used. The predicted value f is defined as the probability summation of the typical fault sizes as follows:

$$f = \sum_{j=1}^{c} a_j p_j \tag{19}$$

where $[p_1, \ldots, p_c]$ is produced by the top softmax layer, which denotes the probability score that each sample belongs to each class size and a_j is the fault size corresponding to the j-th fault size. Sun et al. adopted dual tree complex wavelet transform (DTCWT) to transform raw time-series signal into 2D map, which can approximate shift-invariance and inhibited frequency aliasing and fed the 2D map into CNN models [115]. In [116], an enhanced CNN was proposed for machinery fault diagnosis. To pre-process vibration data, morlet wavelet was used to decompose the vibration signal and obtain wavelet scaleogram. Then, bilinear interpolation was used to rescale the scaleogram into a grayscale image with a size of 32×32. In addition, the adaptation of rectified linear unit and dropout both boosted the model's diagnosis performance. Chen et al. adopted a 2D-CNN for gearbox fault diagnosis, in which the input matrix with a size of 16×16 for CNN was reshaped by a vector containing 256 statistic features including RMS values, standard deviation, skewness, kurtosis, rotation frequency, and applied load [117]. In addition, 11 different structures of CNN were evaluated empirically in their experiments. Weimer et al. did a comprehensive study of various design configurations of deep CNN for visual defect detection [118]. In one specific application: industrial optical inspection, two directions of model configurations including depth (addition of conv-layer) and width (increase of number filters) were investigated. The optimal configuration verified empirically has been presented in Table 2. In [119], CNN was applied in the field of diagnosing the early small faults of front-end controlled wind generator (FSCWG) where the 784×784 input matrix consisted of vibration data of generator input shaft (horizontal) and vibration data of generator output shaft (vertical) in time scale. In [120], You et al. used support vector machine as the classifier on the features extracted by CNN for fault diagnosis of rotating machinery. In Lee's work, they adopted CNN for fault classification and diagnosis in semiconductor manufacturing processes [121]. The 2D input matrix into CNN model was with a processing time axis and a sensor variable axis and the filter was only slid along the processing time axis. The subsequent pooling operation was conducted on the time axis for each feature map. In [122], Wen et al. firstly transformed the input raw time-series signal into 2D image by sampling segments randomly from the raw signal. They fed the 2D image into a very classical 2D CNN structure:

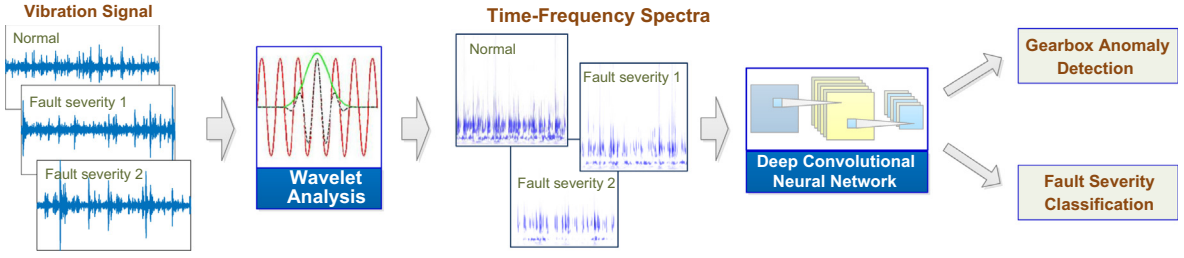

Fig. 8. Illustrations of the proposed 2D-CNN for gearbox fault detection in [108].

Table 2
Summary on configurations of CNN-based MHMS.

	Proposed Models	Configurations of CNN Structures*
2D CNN	*Liu's work [107]*	Input[10 × 512] − 16C[3 × 4] − 16C[3 × 4] − 16P[2 × 2] − 32C[3 × 4] − 32C[3 × 4] − 32P[2 × 2] − FC[∗] − FC[∗]
	Janssens's work [109]	Input[5120 × 2] − 32C[64 × 2] − FC[200]
	Babu's work [111]	Input[27 × 15] − 8C[27 × 4] − 8P[1 × 2] − 14C[1 × 3] − 14P[1 × 2]
	Ding's work [112]	Input[32 × 32] − 20C[7 × 7] − 20P[2 × 2] − 10C[6 × 6] − 10P[2 × 2] − 6P[2 × 2] − FC[185 − 24]
	Guo's work [113]	Input[32 × 32] − 5C[5 × 5] − 5P[2 × 2] − 10C[5 × 5] − 10P[2 × 2] − 10C[2 × 2] − 10P[2 × 2] − FC[100] − FC[50]
	Wang's work [116]	Input[32 × 32] − 64C[3 × 3] − 64P[2 × 2] − 64C[4 × 4] − 64P[2 × 2] − 128C[3 × 3] − 128P[2 × 2] − FC[512]
	Chen's work [117]	Input[16 × 16] − 8C[5 × 5] − 8P[2 × 2]
	Weimer's work [118]	Input[32 × 32] − 64C[3 × 3]2 − 64P[2 × 2] − 128C[3 × 3]3 − 128P[2 × 2] − FC[1024 − 1024]
	Dong's work [119]	Input[784 × 784] − 12C[10 × 10] − 12P[2 × 2] − 24C[10 × 10] − 24P[2 × 2] − FC[200]
	You's work [120]	Input[32 × 32] − 5C[5 × 5] − 5P[2 × 2] − 10C[5 × 5] − 10P[2 × 2] − 10C[2 × 2] − 10P[2 × 2] − FC[100]
1D CNN	*Ince's work [126]*	Input[240] − 60C[9] − 60P[4] − 40C[9] − 40P[4] − 40C[9] − 40P[4] − FC[20]
	Abdeljaber's work [127]	Input[128] − 64C[41] − 64P[2] − 32C[41] − 32P[2] − FC[10 − 10]
	Zhang's work [130]	Input[2048] − 16C[64] − 16P[2] − 4[32C[3] − 32P[2]] − FC[100]
	Sun's work [132]	Input[799] − 100C[200] − 100P[20]

* The symbol *Input*, *C*, *P* and *FC* denote the raw input, convolutional layer, pooling layer and Fully-connected layer, respectively. Here, ∗ denotes the undisclosed hyperparameter in the corresponding paper.

Lenet-5 CNN. Their model achieved very promising results in three different machinery diagnosis tasks including motor bearing fault diagnosis, self-priming centrifugal pump fault diagnosis and axial piston hydraulic pump fault diagnosis. In [123], various CNN trained on different sensors were fused based on an improved Dempster-Shafer theory to derive final prediction. The root mean square maps from Fast Fourier Transformations from sensory data were extracted as the input features for CNN models. In [124], Singh et al. utilized ensemble empirical model decomposition (EEMD) to decompose raw sensory input into intrinsic mode functions (IMF) with selection criteria based on combined model functions (CMF) algorithms, which were used as the input features for CNN models. In [125], short-time Fourier transform, wavelet transform and Hilbert-Huang transformation were adopted to generate image inputs into their CNN model.

As reviewed in Section 2.3, CNN can also be applied to 1D time series signal and the corresponding operations have been elaborated. In [126], the 1D CNN was successfully developed on raw time series data for motor fault detection, in which feature extraction and classification were integrated together. Abdeljaber et al. proposed 1D CNN on normalized vibration signal, which can perform vibration-based damage detection and localization of the structural damage in real-time. The advantage of this approach is its ability to extract optimal damage-sensitive features automatically from the raw acceleration signals, which does not need any additional preprocessing or signal processing approaches [127]. Jing et al. investigated performances of 1D CNN on different data types including raw time data, frequency spectrum data, time–frequency data and several hand-crafted features for gearbox fault detection and CNN achieved the best performance with the feature spectrum [128]. In [129], Zhang et al. proposed CNN with Training Interference (TICNN) for bearing fault diagnosis whose input is raw time-series signal. For data augmentation, kernel with changing dropout rate was applied to the input signal and the batch size was set to be the same value as the number of fault types, which can improve the generalization ability of the trained model. Due to these two modifications, their proposed model was able to achieve high accuracy and stable performance under noisy and varying environment. In [130], Deep Convolutional Neural Networks with Wide First-layer Kernels (WDCNN) was proposed by Zhang et al. The proposed method used raw vibration signals as input (data augmentation was used to generate more inputs), and applied the wide kernels in the first convolutional layer for extracting features and suppressing high frequency noise. Small convolutional kernels in the preceding layers were employed for multi-layer nonlinear mapping. An technique named Adaptive Batch Normalization [131] that parameters in batch normalization were adjusted according to testing samples was implemented to improve the domain adaptation ability of the model.

Different from these previous works where supervised CNNs were adopted, Sun et al. proposed a convolutional discriminative feature learning model to detect induction motor fault diagnosis [132]. As shown in Fig. 9, a feed-forward convolutional pooling architecture was proposed, in which local filters were pre-learned by back-propagation-based neural network (BPNN). Then, the learned representation was fed into SVM for fault conditions classification. Since local filters are learned by BPNN, the following convolutional pooling architecture can extract discriminative and invariant features from the raw vibration data quickly. The input data is 1D vibration signal so that their work also belongs to 1D CNN. In [133], Cabrera et al. adopted convolutional autoencder (CAE) to initialize their supervised CNN model parameters. In CAE, the encoder consists of convolution and max-pooling while the decoder consists of un-pooling as horizontal and vertical replication of activation value and convolution. The training objective of CAE was defined as euclidean distance. In [134], Shao et al. incorporated CNN into DBN by employing convolutional connections in the generative Markov random file structure. In addition, Gaussian visible units were introduced to construct this model. The input into the model was the compressed data learned by autoecnoder as the hidden representations. The softmax classifier was used for bearing fault diagnosis.

In [135], Zhao et al. developed a variant of deep residual networks named as deep residual networks with dynamically weighted wavelet coefficients (DRN + DWWC). The inputs into the model is a series of wavelet packet coefficients on various frequency bands. The DRN consists of several residual building block as a stack of several convolutional layers, batch normal-

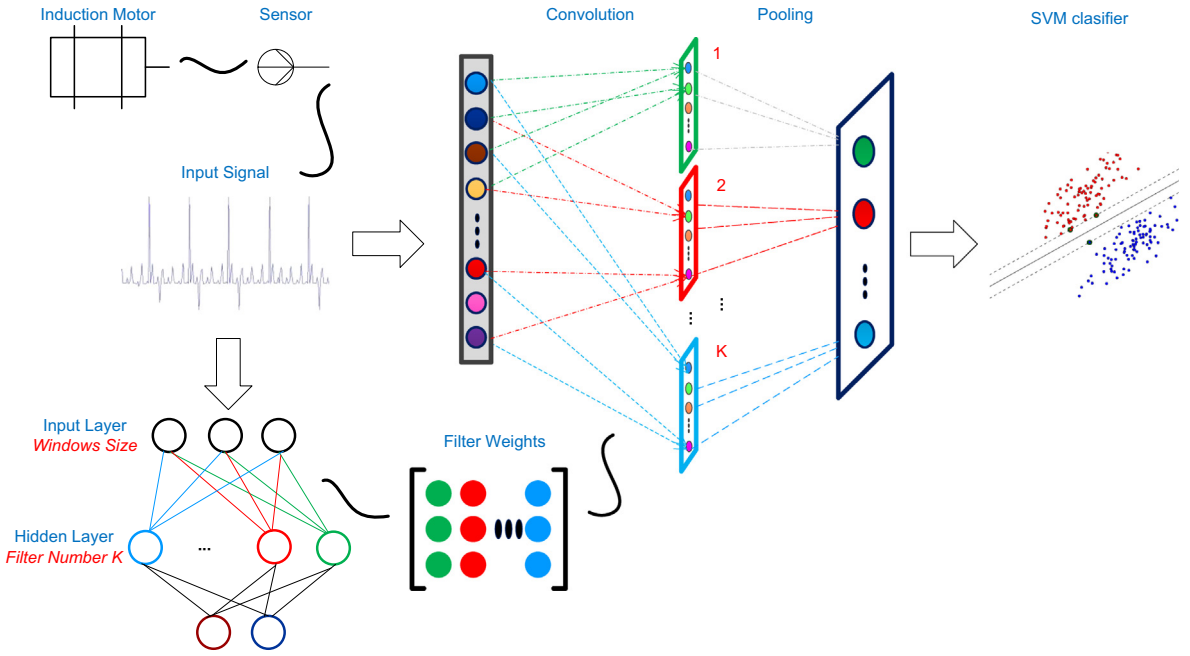

Fig. 9. Illustrations of the proposed unsupervised CNN for induction motor fault diagnosis in [132].

izations (BNs), *relu* activation function and one identity shortcut. In traditional DRN, dynamic weighting layers were designed to apply dynamic weights to the input feature map and elements in each row in feature map shared the same weight. The utilization of dynamic weighting layers focused on emphasizing different contributions of wavelet packet coefficients on different frequency bands. In [136], Pan et al. proposed a novel CNN model named LiftingNet for fault classification, which was based on CNN and Second Generation Wavelet Transform (SGWT). The basic module in LiftingNet consists of split layer, predict layer and update layer. Split layer divided the input sequence into an even series and an odd series. Then, predict and update layers applied circular convolution operation to learn representations of input on different scales via using different kernel sizes. After stacking of the above modules, max-pooling layer and fully-connected layer were applied to learning the final representation. It has been verified that LiftingNet can learn transient, high-frequency component via predict layer and encode gradual, low-frequency component via update layer. In addition, the large-size kernels and nonlinear functions were able to filter noise.

To present an overview about all these above CNN models that have been successfully applied in the area of MHMS, their architectures have been summarized in Table 2. To explain the used abbreviation, the structure of CNN applied in Weimer's work [118] is denoted as Input$[32 \times 32] - 64C[3 \times 3]2 - 64P[2 \times 2] - 128C[3 \times 3]3 - 128P[2 \times 2] - FC[1024 - 1024]2$. It means the input 2D data is 32×32 and the CNN firstly applies 2 convolutional layers with the same design that the number of filters is 64 and the filter size is 3×3, then stackes one max-pooling layer whose pooling size is 2×2, then applies 3 convolutional layers with the same design that the number of filters is 128 and the filer size is 3×3, then applies a pooling layer whose pooling size is 2×2, and finally adopts two fully-connected layers whose hidden neuron numbers are both 1024. It should be noted that the size of output layer is not given here, considering it is task-specific and usually set to be the number of categories.

3.4. RNN for machine health monitoring

The majority of machinery data belong to sensor data, which are in nature time series. RNN models including LSTM and GRU have emerged as one kind of popular architectures to handle sequential data with its ability to encode temporal information. These advanced RNN models have been proposed to relief the difficulty of training behind vanilla RNN for machine health monitoring recently. In [137], Yuan et al. investigated three RNN models including vanilla RNN, LSTM and GRU models for fault diagnosis and prognostics of aero engine. They found these advanced RNN models of LSTM and GRU outperformed vanilla RNN. Another interesting observation was the ensemble model of the above three RNN variants did not boost the performance of LSTM. Zhao et al. presented an empirical evaluation of LSTMs-based machine health monitoring system in the tool wear test [138]. The applied LSTM model encoded the raw sensory data into vectors and predicted the corresponding tool wear. Zhao et al. further designed a more complex deep learning model combining CNN and LSTM named Convolutional Bi-directional Long Short-Term Memory Networks (CBLSTM) [139]. As shown in Fig. 10, CNN was used to extract robust local features from the sequential input, and then bi-directional LSTM was adopted to encode temporal information on the

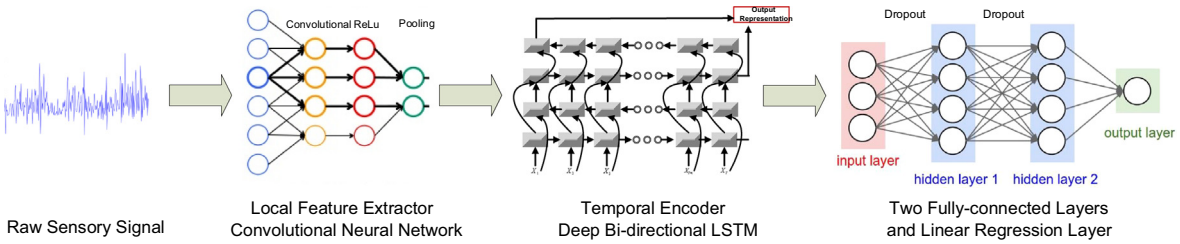

Fig. 10. Illustrations of the proposed Convolutional Bi-directional Long Short-Term Memory Networks in [139].

sequential output of CNN. Stacked fully-connected layers and linear regression layer were finally added to predict the target value. In tool wear test, the proposed model was able to outperform several state-of-the-art baseline methods including conventional LSTM models. Different from the previous automatic feature learning models, Zhao et al. proposed a hybrid approach that combines handcrafted feature design with automatic feature learning for machine health monitoring [140]. As illustrated in Fig. 11, features from windows of input time series were extracted and fed into an enhanced bi-directional GRU network. The enhanced bi-directional GRU network consists of two modules including bi-directional GRU and weighted feature averaging. Their proposed model in three machine health monitoring tasks: tool wear prediction, gearbox fault diagnosis and incipient bearing fault detection has shown the effectiveness and generalization of the proposed hybrid approach combing handcrafted feature design and automatic feature learning. In [141], Malhotra proposed a very interesting structure for RUL prediction. They designed a LSTM-based encoder-decoder structure, which LSTM-based encoder firstly transformed a multivariate input sequence to a fixed-length vector and then, LSTM decoder used vectors to produce the target sequence. When it comes to RUL prediction, their assumptions lies that the model can be firstly trained in raw signal corresponding to normal behavior in an unsupervised way. Then, the reconstruction error can be used to compute

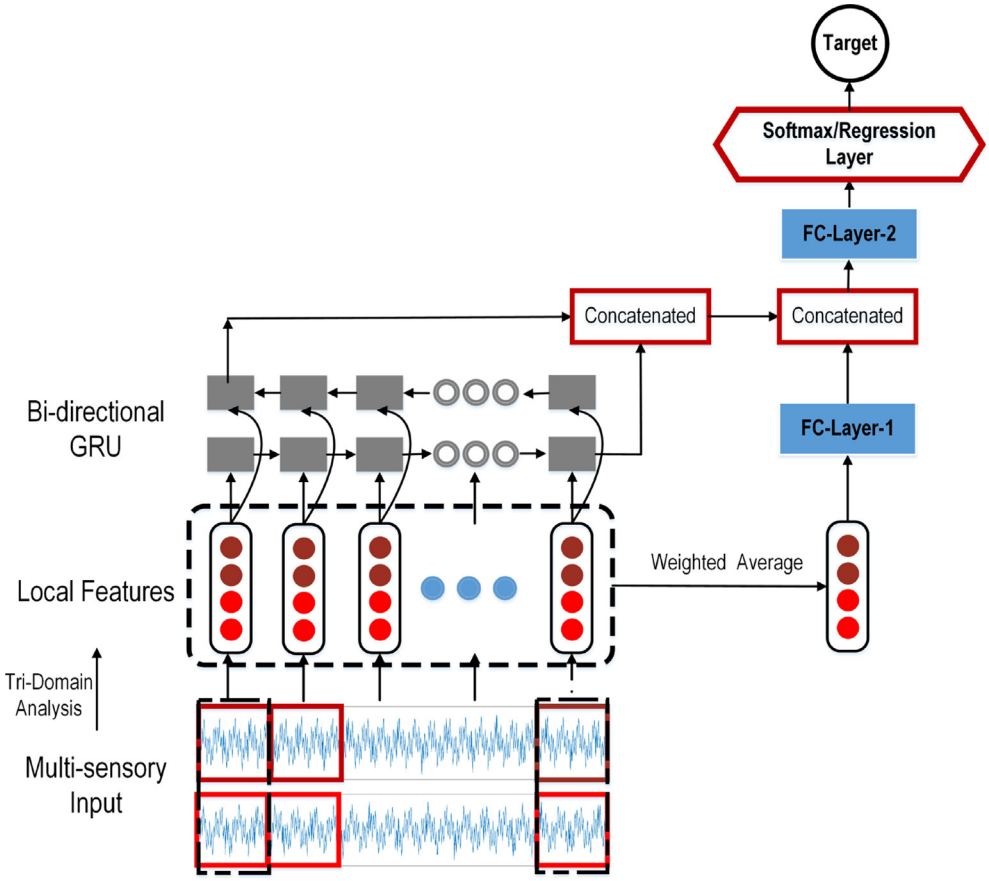

Fig. 11. Illustrations of the proposed Local Feature-based Gated Recurrent Unit Networks in [140].

health index (HI), which is then used for RUL estimation. It is intuitive that the large reconstruction error corresponds to an unhealthy machine condition.

3.5. Fault feature extraction for DL-based MHMS

In the above sessions, the applications of various deep learning methods in machine health monitoring systems have been reviewed. It can be found that feature extraction or feature preprocessing are required in certain works. Although deep learning can learn representations from scratch, the noisy sensory data acquired from machines, the complexity in the machinery operating systems and the insufficient data samples all make it is necessary to apply fault feature extraction before raw data are fed into DL models. In addition, rare and inconsistent sensory signal, i.e., outliers, may have some unpredictable influence on the nonlinear and real mechanical system [142–144]. The appropriate fault feature extraction can alleviate this negative effect of outliers. Therefore, a brief summary of these feature extraction techniques adopted in the above works is given following several aspects:

3.5.1. Time domain

Raw sensory data is in nature time-series signal. Therefore, some statistical time-domain features could be extracted as discriminative features fed into DL systems. Mean, root mean square (RMS), standard deviation and variances were used a lot. Due to the stationary signal, skewness and kurtosis were also extracted as input features.

3.5.2. Frequency domain

Due to the existence of periodical impulse in several machinery faults, their dominant frequency components are informative and discriminative features. FFT (Fast Fourier Transform) can be applied to transforming the time-domain vibration signals into frequency-domain ones. Mean frequency, root variance frequency, spectral skewness and spectral kurtosis as first-order, second-order, third-order and fourth-order moments of the Fourier spectrum can be calculated as features. It should be noted that frequency domain representations are commonly used for stationary signal.

3.5.3. Time-frequency domain

Time-frequency domain features are useful for non-stationary signal. Short-time Fourier Transform, wavlet transform/decomposition and empricial model decomposition (EMD) are widely used to convert the one-dimensional signal into two-dimensional signal of time and frequency.

In addition, we provided implementations of several feature extraction methods in these three above domains,[1] which will be illustrated in the following section with more details.

4. Experimental evaluations of various DL techniques

In this section, a systematic evaluation of these above discussed deep learning models on machine health monitoring tasks is provided. In particular, the tool wear sensing task has been introduced.

4.1. Implementation details

Dataset description: dataset were sampled from a high speed CNC machine during dry milling operations and its schematic diagram of experimental platform is shown in 12. The detailed experimental settings can be found in [145], where seven sensors including force and vibration ones in three directions and AE-RMS have been placed. The ground-truth value were obtained by using a LEICA MZ12 microscope to measure each individual flute after finishing each surface, i.e., each cut number. Machine learning models are adopted to predict the actual flank wear from the sensory data. Three individual cutter records denoted as c1, c4 and c6 are available and each record contains 315 data samples.

In our experiments, c4 is used as testing data while the other records c1 and c6 are used as training data. Considering the high dimensionality of the raw time series signal, feature extractions are applied firstly. Seven kinds of features including time domain, frequency domain and time–frequency domain are designed, which could be found in Table 3. Here, the wavelet energy feature is the energy of a 8-level wavelet packet decomposition using db1, which corresponds to the wavelet coefficient with higher energy that is related to the characteristic frequency of the machine. Considering seven sensors were deployed, the dimensionality of the hand-crafted feature vector is 70. For LSTM and CNN, the input data is tensor so that the data is sliced into 20 windows and then fed into feature extraction, respectively. For other models such as SVM, the input data is vector so that the whole time series is fed into feature extraction. Therefore, we have two kinds of extracted features, one is in shape of 20*70 and the other is in shape of 70.

Compared Approaches: These following methods are compared:

[1] https://github.com/ClockworkBunny/MHMS_DEEPLEARNING/blob/master/code/feature_extract.py.

* *Linear SVR*: Linear Support Vector Regression, whose input features are extracted features of the whole time series. The regularization term is set to 1.
* *RBF SVR*: Support Vector Regression with RBF kernel, whose input features are extracted features of the whole time series. The regularization term is set to 1.
* *Random Forest*: Random Forest Regressor whose input features are extracted features of the whole time series. The number of estimator is set to 50 and the max depth of decision tree is set to 2.
* *Neural Network*: neural network whose input features are extracted features of the whole time series. The neural network contains two hidden layers, whose sizes are 70 and 140, respectively. To prevent overfiting, the dropout layer with a masking probability of 0.2 is applied on the last layer.

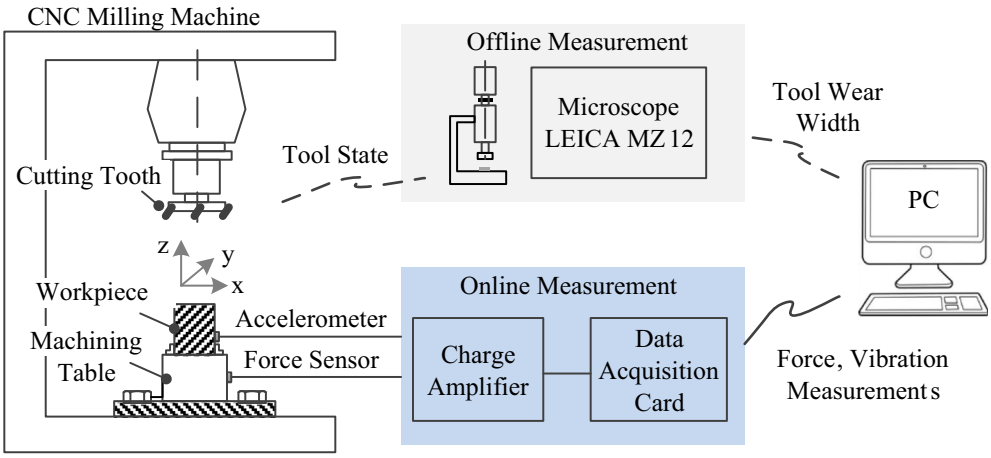

Fig. 12. Schematic of the experimental setup for tool wear prediction [23].

Table 3
List of Extracted Features.

Domain	Features	Expression
Statistical	RMS	$z_{rms} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} z_i^2}$
	Variance	$z_{var} = \frac{1}{n}\sum_{i=1}^{n}(z_i - \bar{z})^2$
	Maximum	$z_{max} = max(z)$
	Skewness	$z_{skew} = E[(\frac{z-\mu}{\sigma})^3]$
	Kurtosis	$z_{kurt} = E[(\frac{z-\mu}{\sigma})^4]$
	Peak-to-Peak	$z_{p-p} = max(z) - min(z)$
Frequency	Spectral Skewness	$f_{skew} = \sum_{i=1} k(\frac{f_i - \bar{f}}{\sigma})^3 S(f_i)$
	Spectral Kurtosis	$f_{kurt} = \sum_{i=1} k(\frac{f_i - \bar{f}}{\sigma})^4 S(f_i)$
Time–Frequency	Wavelet Energy	$E_{WT} = \sum_{i=1}^{N} wt_{\phi}^2(i)/N$

Table 4
The Results of all the approaches in tool wear prediction under the two criteria including MAE and MSE.

Method	MAE	MSE
Linear SVR	13.7	248.9
RBF SVR	17.6	402.9
Random Forest	14.4 ± 0.2	289 ± 5.9
Neural Network	11.5 ± 1.2	191.6 ± 26.8
Auto-encoder	11.2 ± 1.9	185 ± 41.9
Denoising Auto-encoder	9.3 ± 1.8	143.5 ± 40.4
DBN	11.7 ± 1.3	194.6 ± 34.2
LSTM	11.0 ± 2.5	193.9 ± 65.3
Bi-directional LSTM	10.6 ± 1.9	191.5 ± 56.1
CNN	11.0 ± 1.3	197.2 ± 30.9

* *Auto-encoder*: auto-encoder whose input features are extracted features of the whole time series. The loss function of pre-training is mean-squared error. The sizes of hidden layers in pre-training are 100 and 140, respectively. In supervised training, a layer with a size of 900 is added following these two hidden layers.
* *Denoising-Auto-encoder*: denoising auto-encoder whose input features are extracted features of the whole time series. Compared to normal auto-encoder, a dropout noise is added on the input features with a masking probability of 0.01.

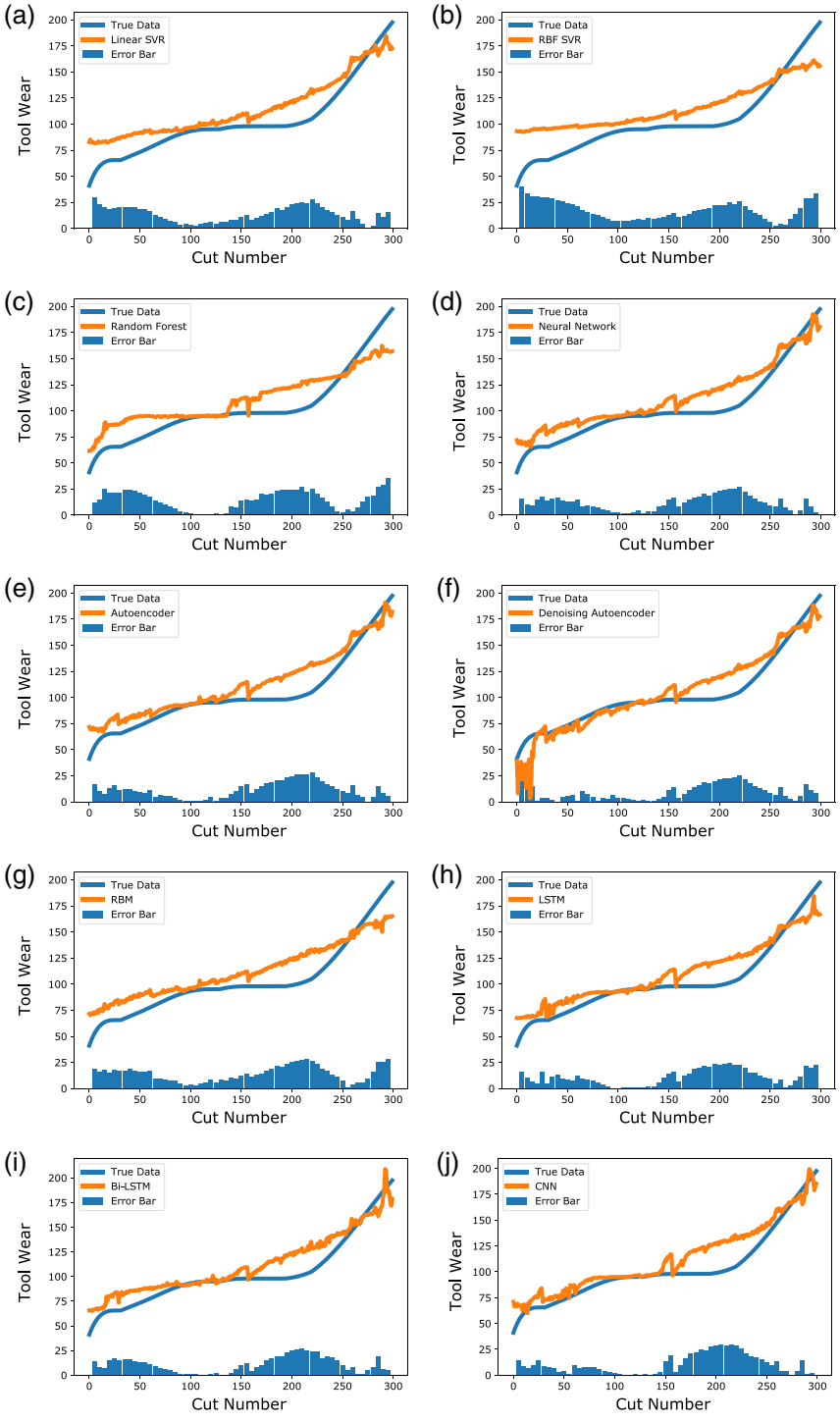

Fig. 13. Regression Result of All Compared Methods. (a) Linear SVR, (b) RBF SVR, (c) Random Forest, (d) Neural Network, (e) Autoencoder, (f) Denoising Autoencoder, (g) DBN, (h) LSTM, (i) Bi-LSTM, (j) CNN.

* *DBN*: deep belief network whose input features are extracted features of the whole time series. The hidden layer sizes are 100 and 140, respectively.
* *CNN*: Convolutional Neural Network whose input features are sequential features. Two 1D convolutional layers with windows sizes of 3 are adopted. And their hidden layer sizes are set to 100 and 140. Then, one max-pooling layer is added, which is followed by a fully-connected layer with a size of 900 and a dropout layer whose dropout probability is 0.2.
* *LSTM*: Long-short Term Memory Network whose input features are sequential features. We stacked two recurrent layers whose hidden layer sizes are set to 100 and 140. A fully-connected layer with a size of 900 and a dropout layer with a masking probability of 0.2 are added.
* *Bi-directional LSTM*: Bi-directional Long-short Term Memory Network whose input features are sequential features. Compared to LSTM, the data are fed into bi-directional LSTM in two directions: from beginning to end and from end to beginning.

In addition, the dataset and code have been published[2]. Due to the privacy issue and potential copyright concern, we only provide the extracted features for these data instead of raw time-series. Since almost all deep learning models require random parameter initialization, all the comparative models were run five times. Here, we adopt two measures including mean absolute error (MAE) and mean squared error (MSE).

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\tilde{y}_i - y_i| \tag{20}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\tilde{y}_i - y_i)^2} \tag{21}$$

where y_i and $\tilde{y}_i$ are true and predicted too wear depth.

4.2. Experimental results

The performances of all compared methods are shown in Table 4. In addition, the regression results are all visualized in Fig. 13. It should be noted that the hyper-parameters of all models are set to be default values without fine-tuning. In this section, we are attempting to shed some lights over the application of deep learning models on machine health monitoring tasks. It should be noted that, due to the default setting of hyper-parameter selection and the small data size, the improvement of deep learning models compared to traditional methods is not so huge. However, as shown in Table 4, it still could be found that deep learning models perform better than these conventional machine learning models. In addition, due to pre-training, auto-encoder is able to achieve better performance than neural network. However, DBN performed slightly worse than neural network, which may be explained by the fact that DBN is hard to train due to sampling required at each layer. The introduction of dropout noise further improve the performances of denoising-autoencoders. Although CNN, LSTM and Bi-LSTM all perform slightly worse than denoising autoencoder, hyperparameter selection may boost their performances considering the complexity of these models. But bi-directional LSTM performs better than LSTM. It means that the bi-directional processing of time-series signal is suitable for machine health monitoring.

* Compared to conventional machine learning models, deep learning models are able to achieve superior performances in the field of machine health monitoring.
* Pre-training in autoencoder can boost the performance of machine health monitoring.
* Denoising technique is beneficial for machine health monitoring.
* CNN, LSTM and their variants can handle machine health monitoring. However, due to the model complexity, hyperparameter selection is required to achieve state-of-the-art performances.

5. Summary and future directions

In this paper, we have provided a systematic overview of the state-of-the-art DL-based MHMS. Deep learning, as a subfield of machine learning, is serving as a bridge between big machinery data and data-driven MHMS. Therefore, within the past five years, they have been applied in various machine health monitoring tasks. These proposed DL-based MHMS are summarized according to four categories of DL architecture as: Auto-encoder models, Restricted Boltzmann Machines models, Convolutional Neural Networks and Recurrent Neural Networks. Since the momentum of the research of DL-based MHMS is growing fast, we hope the messages about the capabilities of these DL techniques, especially representation learning for complex machinery data and target prediction for various machine health monitoring tasks, can be conveyed to readers. Through these previous works, it can be found that DL-based MHMS do not require extensive human labor and expert knowledge, i.e., the end-to-end structure is able to map raw machinery data to targets. Therefore, the application of deep

[2] Please find code and data inhttps://github.com/ClockworkBunny/MHMS_DEEPLEARNING.

learning models are not restricted to specific kinds of machines, which can be a general solution to address the machine health monitoring problems. Besides, some research trends and potential future research directions are given as follows:

* *Open-source Large Dataset*: Due to the huge model complexity behind DL methods, the performance of DL-based MHMS heavily depends on the scale and quality of datasets. On the other hand, the depth of DL model is limited by the scale of datasets. As a result, the benchmark CNN model for image recognition has 152 layers, which can be supported by the large dataset ImageNet containing over ten million annotated images [146,147]. In contrast, the proposed DL models for MHMS may stack up to 5 hidden layers. And the model trained in such kind of large datasets can be the model initialization for the following specific task/dataset. Therefore, it is meaningful to design and publish large-scale machinery datasets.
* *Utilization of Domain Knowledge*: deep learning is not a skeleton key to all machine health monitoring problems. Domain knowledge can contribute to the success of applying DL models on machine health monitoring. For example, extracting discriminative features can reduce the size of the followed DL models and appropriate task-specific regularization term can boost the final performance [84].
* *Model and Data Visualization*: deep learning techniques, especially deep neural networks, have been regarded as black boxes models, i.e., their inner computation mechanisms are unexplainable. Visualization of the learned representation and the applied model can offer some insights into these DL models, and then these insights achieved by this kind of interaction can facilitate the building and configuration of DL models for complex machine health monitoring problems. Some visualization techniques have been proposed including t-SNE model for high dimensional data visualization [148] and visualization of the activations produced by each layer and features at each layer of a DNN via regularized optimization [149].
* *Transferred Deep Learning*: Transfer learning tries to apply knowledge learned in one domain to a different but related domain [25]. This research direction is meaningful in machine health monitoring, since some machine health monitoring problems have sufficient training data while other areas lack training data. The machine learning models including DL models trained in one domain can be transferred to the other domain. Some previous works focusing on transferred feature extraction/dimensionality reduction have been done [150,151]. In [152], a Maximum Mean Discrepancy (MMD) measure evaluating the discrepancy between source and target domains was added into the target function of deep neural networks.
* *Imbalanced Class*: The class distribution of machinery data in real life normally follows a highly-skewed one, in which most data samples belong to few categories. For example, the number of fault data is much less than the one of health data in fault diagnosis. Some enhanced machine learning models including SVM and ELM have been proposed to address this imbalanced issue in machine health monitoring [153,154]. Recently, some interesting methods investigating the application of deep learning in imbalanced class problems have been developed, including CNN models with class resampling or cost-sensitive training [155] and the integration of boot strapping methods and CNN model [156].

It is believed that deep learning will have a more and more prospective future impacting machine health monitoring, especially in the age of big machinery data.

Acknowledgment

This work has been supported in part by the National Natural Science Foundation of China (51575102).

References

[1] S. Yin, X. Li, H. Gao, O. Kaynak, Data-based techniques focused on modern industry: an overview, IEEE Trans. Industr. Electron. 62 (1) (2015) 657–667, ISSN 0278-0046.
[2] S. Jeschke, C. Brecher, H. Song, D.B. Rawat, Industrial Internet of Things, Springer, 2017.
[3] D. Lund, C. MacGillivray, V. Turner, M. Morales, Worldwide and regional internet of things (iot) 2014–2020 forecast: A virtuous circle of proven value and demand, International Data Corporation (IDC), Tech. Rep.
[4] Y. Li, T. Kurfess, S. Liang, Stochastic prognostics for rolling element bearings, Mech. Syst. Signal Process. 14 (5) (2000) 747–762.
[5] C.H. Oppenheimer, K.A. Loparo, Physically based diagnosis and prognosis of cracked rotor shafts, AeroSense 2002, International Society for Optics and Photonics, 2002, pp. 122–132.
[6] M. Yu, D. Wang, M. Luo, Model-based prognosis for hybrid systems with mode-dependent degradation behaviors, IEEE Trans. Industr. Electron. 61 (1) (2014) 546–554.
[7] A.K. Jardine, D. Lin, D. Banjevic, A review on machinery diagnostics and prognostics implementing condition-based maintenance, Mech. Syst. Signal Processing 20 (7) (2006) 1483–1510.
[8] V. Stojanovic, N. Nedic, D. Prsic, L. Dubonjic, V. Djordjevic, Application of cuckoo search algorithm to constrained control problem of a parallel robot platform, Int. J. Adv. Manuf. Technol. 87 (9–12) (2016) 2497–2507.
[9] D. Pršić, N. Nedić, V. Stojanović, A nature inspired optimal control of pneumatic-driven parallel robot platform, Proc. Inst. Mech. Eng., Part C: J. Mech. Eng. Sci. 231 (1) (2017) 59–71.
[10] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: towards real-time object detection with region proposal networks, in: Advances in Neural Information Processing Systems, 91–99, 2015.
[11] R. Collobert, J. Weston, A unified architecture for natural language processing: Deep neural networks with multitask learning, in: Proceedings of the 25th International Conference on Machine learning, ACM, 160–167, 2008.
[12] G. Hinton, L. Deng, D. Yu, G.E. Dahl, A.-R. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T.N. Sainath, et al, Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups, IEEE Signal Process. Mag. 29 (6) (2012) 82–97.

[13] M.K. Leung, H.Y. Xiong, L.J. Lee, B.J. Frey, Deep learning of the tissue-regulated splicing code, Bioinformatics 30 (12) (2014) i121–i129.
[14] J. Schmidhuber, Deep Learning in Neural Networks: An Overview, Neural Networks 61 (2015) 85–117, doi: 10.1016/j.neunet.2014.09.003, published online 2014; based on TR arXiv:1404.7828 [cs.NE].
[15] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436–444.
[16] R. Raina, A. Madhavan, A.Y. Ng, Large-scale deep unsupervised learning using graphics processors, in: Proceedings of the 26th Annual International Conference on Machine Learning, ACM, 873–880, 2009.
[17] G.E. Hinton, Learning multiple layers of representation, Trends Cognitive Sci. 11 (10) (2007) 428–434.
[18] A. Widodo, B.-S. Yang, Support vector machine in machine condition monitoring and fault diagnosis, Mech. Syst. Signal Processing 21 (6) (2007) 2560–2574.
[19] J. Yan, J. Lee, Degradation assessment and fault modes classification using logistic regression, J. Manuf. Sci. Eng. 127 (4) (2005) 912–914.
[20] V. Muralidharan, V. Sugumaran, A comparative study of Naïve Bayes classifier and Bayes net classifier for fault diagnosis of monoblock centrifugal pump using wavelet analysis, Appl. Soft Comput. 12 (8) (2012) 2023–2029.
[21] Y. Bengio, A. Courville, P. Vincent, Representation learning: Aa review and new perspectives, IEEE Trans. Pattern Anal. Mach. Intell. 35 (8) (2013) 1798–1828.
[22] A. Malhi, R.X. Gao, PCA-based feature selection scheme for machine defect classification, IEEE Trans. Instrum. Meas. 53 (6) (2004) 1517–1525.
[23] J. Wang, J. Xie, R. Zhao, L. Zhang, L. Duan, Multisensory fusion based virtual tool wear sensing for ubiquitous manufacturing, Robotics Computer-Integrated Manuf. 45 (2017) 47–58.
[24] J. Wang, J. Xie, R. Zhao, K. Mao, L. Zhang, A new probabilistic kernel factor analysis for multisensory data fusion: application to tool condition monitoring, IEEE Trans. Instrum. Meas. 65 (11) (2016) 2527–2537, ISSN 0018-9456.
[25] S.J. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 22 (10) (2010) 1345–1359.
[26] P. Vincent, H. Larochelle, Y. Bengio, P.-A. Manzagol, Extracting and composing robust features with denoising autoencoders, in: Proceedings of the 25th International Conference on Machine Learning, ACM, 1096–1103, 2008.
[27] G.E. Hinton, S. Osindero, Y.-W. Teh, A fast learning algorithm for deep belief nets, Neural Comput. 18 (7) (2006) 1527–1554.
[28] R. Salakhutdinov, G.E. Hinton, Deep boltzmann machines., in: AISTATS, vol. 1, 3, 2009.
[29] P. Sermanet, S. Chintala, Y. LeCun, Convolutional neural networks applied to house numbers digit classification, in: Pattern Recognition (ICPR), 2012 21st International Conference on, IEEE, 3288–3291, 2012.
[30] K.-I. Funahashi, Y. Nakamura, Approximation of dynamical systems by continuous time recurrent neural networks, Neural Networks 6 (6) (1993) 801–806.
[31] L. Deng, D. Yu, Deep learning: methods and applications, Found. Trends Signal Process. 7 (2014) 197–387.
[32] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016.
[33] A. Ng, Sparse autoencoder, CS294A Lecture Notes 72 (2011) 1–19.
[34] G. Hinton, A practical guide to training restricted Boltzmann machines, Momentum 9 (1) (2010) 926.
[35] A.-R. Mohamed, G.E. Dahl, G. Hinton, Acoustic modeling using deep belief networks, IEEE Trans. Audio, Speech, Language Processing 20 (1) (2012) 14–22.
[36] B.B. Le Cun, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard, L.D. Jackel, Handwritten digit recognition with a back-propagation network, in: Advances in Neural Information Processing Systems, Citeseer, 1990.
[37] K. Jarrett, K. Kavukcuoglu, Y. Lecun, et al., What is the best multi-stage architecture for object recognition?, in: 2009 IEEE 12th International Conference on Computer Vision, IEEE, 2146–2153, 2009.
[38] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, 1097–1105, 2012.
[39] O. Abdel-Hamid, A.-R. Mohamed, H. Jiang, G. Penn, Applying convolutional neural networks concepts to hybrid NN-HMM model for speech recognition, in: 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 4277–4280, 2012.
[40] Y. Kim, Convolutional neural networks for sentence classification, arXiv preprint arXiv:1408.5882.
[41] H. Jaeger, Tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the "echo state network" approach, GMD-Forschungszentrum Informationstechnik, 2002.
[42] C.L. Giles, C.B. Miller, D. Chen, H.-H. Chen, G.-Z. Sun, Y.-C. Lee, Learning and extracting finite state automata with second-order recurrent neural networks, Neural Comput. 4 (3) (1992) 393–405.
[43] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780.
[44] F.A. Gers, J. Schmidhuber, F. Cummins, Learning to forget: continual prediction with LSTM, Neural Comput. 12 (10) (2000) 2451–2471.
[45] F.A. Gers, N.N. Schraudolph, J. Schmidhuber, Learning precise timing with LSTM recurrent networks, J. Mach. Learn. Res. 3 (2002) 115–143.
[46] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using RNN encoder-decoder for statistical machine translation, arXiv preprint arXiv:1406.1078.
[47] J. Chung, C. Gulcehre, K. Cho, Y. Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling, arXiv preprint arXiv:1412.3555.
[48] S. Ruder, An overview of gradient descent optimization algorithms, arXiv preprint arXiv:1609.04747.
[49] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in: Proceedings of the thirteenth international conference on artificial intelligence and statistics, 249–256, 2010.
[50] H. Su, K.T. Chong, Induction machine condition monitoring using neural network modeling, IEEE Trans. Ind. Electron. 54 (1) (2007) 241–249, ISSN 0278-0046.
[51] B. Li, M.-Y. Chow, Y. Tipsuwan, J.C. Hung, Neural-network-based motor rolling bearing fault diagnosis, IEEE Trans. Industr. Electron. 47 (5) (2000) 1060–1069.
[52] B. Samanta, K. Al-Balushi, Artificial neural network based fault diagnostics of rolling element bearings using time-domain features, Mech. Syst. Signal Processing 17 (2) (2003) 317–328.
[53] M. Aminian, F. Aminian, Neural-network based analog-circuit fault diagnosis using wavelet transform as preprocessor, IEEE Trans. Circuits Syst. II: Analog Digital Signal Processing 47 (2) (2000) 151–156.
[54] W. Sun, S. Shao, R. Zhao, R. Yan, X. Zhang, X. Chen, A sparse auto-encoder-based deep neural network approach for induction motor faults classification, Measurement 89 (2016) 171–178.
[55] C. Lu, Z.-Y. Wang, W.-L. Qin, J. Ma, Fault diagnosis of rotary machinery components using a stacked denoising autoencoder-based health state identification, Signal Processing 130 (2017) 377–388.
[56] S. Tao, T. Zhang, J. Yang, X. Wang, W. Lu, Bearing fault diagnosis method based on stacked autoencoder and softmax regression, in: Control Conference (CCC), 2015 34th Chinese, IEEE, 6331–6335, 2015.
[57] F. Jia, Y. Lei, J. Lin, X. Zhou, N. Lu, Deep neural networks: A promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data, Mech. Syst. Signal Processing 72 (2016) 303–315.
[58] J. Sun, C. Yan, J. Wen, Intelligent bearing fault diagnosis method combining compressed data acquisition and deep learning, IEEE Trans. Instrum. Meas. 67 (1) (2018) 185–195.
[59] F. Zhou, Y. Gao, C. Wen, A novel multimode fault classification method based on deep learning, J. Control Sci. Eng. (2017) 14, Article ID 3583610.
[60] T. Junbo, L. Weining, A. Juneng, W. Xueqian, Fault diagnosis method study in roller bearing based on wavelet transform and stacked auto-encoder, in: The 27th Chinese Control and Decision Conference (2015 CCDC), IEEE, 4608–4613, 2015.

[61] Z. Huijie, R. Ting, W. Xinqing, Z. You, F. Husheng, Fault diagnosis of hydraulic pump based on stacked autoencoders, in: 2015 12th IEEE International Conference on Electronic Measurement Instruments (ICEMI), vol. 1, 58–62, 2015.
[62] H. Liu, L. Li, J. Ma, Rolling Bearing Fault Diagnosis Based on STFT-Deep Learning and Sound Signals (2016), Shock and Vibration ArticleID 6127479, 12 pages.
[63] G.S. Galloway, V.M. Catterson, T. Fay, A. Robb, C. Love, Diagnosis of tidal turbine vibration data through deep neural networks, in: Proceedings of the 3rd European Conference of the Prognostic and Health Management Society, PHM Society, 172–180, 2016.
[64] K. Li, Q. Wang, Study on signal recognition and diagnosis for spacecraft based on deep learning method, in: Prognostics and System Health Management Conference (PHM), IEEE, 1–5, 2015.
[65] L. Guo, H. Gao, H. Huang, X. He, S. Li, Multifeatures Fusion and Nonlinear Dimension Reduction for Intelligent Bearing Condition Monitoring (2016), Shock and Vibration ArticleID 4632562, 10pages.
[66] N.K. Verma, V.K. Gupta, M. Sharma, R.K. Sevakula, Intelligent condition based monitoring of rotating machines using sparse auto-encoders, in: 2013 IEEE Conference on Prognostics and Health Management (PHM), 1–7, 2013.
[67] M. Sohaib, C.-H. Kim, J.-M. Kim, A hybrid feature model and deep-learning-based bearing fault diagnosis, Sensors 17 (12) (2017) 2876.
[68] F. Cheng, J. Wang, L. Qu, W. Qiao, Rotor current-based fault diagnosis for DFIG wind turbine drivetrain gearboxes using frequency analysis and a deep classifier, IEEE Trans. Ind. Appl. (2017).
[69] R. Chen, S. Chen, M. He, D. He, B. Tang, Rolling bearing fault severity identification using deep sparse auto-encoder network with noise added sample expansion, Proc. Inst. Mech. Eng., Part O: J. Risk Reliab. 231 (6) (2017) 666–679.
[70] V.V. Kishore K. Reddy, Soumalya Sarkar, M. Giering, Anomaly Detection and Fault Disambiguation in Large Flight Data: A Multi-modal Deep Auto-encoder Approach, in: Annual Conference of the Prognostics and Health Management Society, Denver, Colorado, 1–8, 2016.
[71] Z. Chen, W. Li, Multisensor feature fusion for bearing fault diagnosis using sparse autoencoder and deep belief network, IEEE Trans. Instrum. Meas. 66 (7) (2017) 1693–1702.
[72] R. Thirukovalluru, S. Dixit, R.K. Sevakula, N.K. Verma, A. Salour, Generating feature sets for fault diagnosis using denoising stacked auto-encoder, in: IEEE International Conference on Prognostics and Health Management (ICPHM), 1–7, 2016.
[73] L. Wang, X. Zhao, J. Pei, G. Tang, Transformer fault diagnosis using continuous sparse autoencoder, SpringerPlus 5 (1) (2016) 1–13.
[74] W. Mao, J. He, Y. Li, Y. Yan, Bearing fault diagnosis with auto-encoder extreme learning machine: A comparative study, Proc. Inst. Mech. Eng., Part C: J. Mech. Eng. Sci. 231 (8) (2017) 1560–1578.
[75] E. Cambria, G.B. Huang, L.L.C. Kasun, H. Zhou, C.M. Vong, J. Lin, J. Yin, Z. Cai, Q. Liu, K. Li, V.C.M. Leung, L. Feng, Y.S. Ong, M.H. Lim, A. Akusok, A. Lendasse, F. Corona, R. Nian, Y. Miche, P. Gastaldo, R. Zunino, S. Decherchi, X. Yang, K. Mao, B.S. Oh, J. Jeon, K.A. Toh, A.B.J. Teoh, J. Kim, H. Yu, Y. Chen, J. Liu, Extreme learning machines [Trends Controversies], IEEE Intell. Syst. 28 (6) (2013) 30–59.
[76] F. Jia, Y. Lei, L. Guo, J. Lin, S. Xing, A neural network constructed by deep learning technique and its application to intelligent fault diagnosis of machines, Neurocomputing 272 (2018) 619–628.
[77] H. Shao, H. Jiang, H. Zhao, F. Wang, A novel deep autoencoder feature learning method for rotating machinery fault diagnosis, Mech. Syst. Signal Processing 95 (2017) 187–204.
[78] H. Shao, H. Jiang, Y. Lin, X. Li, A novel method for intelligent fault diagnosis of rolling bearings using ensemble deep auto-encoders, Mech. Syst. Signal Processing 102 (2018) 278–297.
[79] H. Shao, H. Jiang, F. Wang, H. Zhao, An enhancement deep feature fusion method for rotating machinery fault diagnosis, Knowl.-Based Syst. 119 (2017) 200–220.
[80] C. Li, W. Zhang, G. Peng, S. Liu, Bearing fault diagnosis using fully-connected winner-take-all autoencoder, IEEE Access (2017), https://doi.org/10.1109/ACCESS.2017.2717492.
[81] W. Lu, X. Wang, C. Yang, T. Zhang, A novel feature extraction method using deep neural network for rolling bearing fault diagnosis, in: The 27th Chinese Control and Decision Conference (2015 CCDC), 2427–2431, 2015.
[82] J. Deutsch, D. He, Using deep learning based approaches for bearing remaining useful life prediction, in: Annual Conference of the Prognostics and Health Management Society, PHM Society, 1–7, 2016.
[83] J. Deutsch, D. He, Using deep learning-based approach to predict remaining useful life of rotating components, IEEE Trans. Syst., Man, Cybern.: Syst. 48 (1) (2018) 11–20.
[84] L. Liao, W. Jin, R. Pavel, Enhanced restricted boltzmann machine with prognosability regularization for prognostics and health assessment, IEEE Trans. Industr. Electron. 63 (11) (2016) 7076–7083.
[85] C. Li, R.-V. Sanchez, G. Zurita, M. Cerrada, D. Cabrera, R.E. Vásquez, Multimodal deep support vector classification with homologous features and its application to gearbox fault diagnosis, Neurocomputing 168 (2015) 119–127.
[86] C. Li, R.-V. Sánchez, G. Zurita, M. Cerrada, D. Cabrera, Fault diagnosis for rotating machinery using vibration measurement deep statistical feature learning, Sensors 16 (6) (2016) 895.
[87] C. Li, R.-V. Sanchez, G. Zurita, M. Cerrada, D. Cabrera, R.E. Vásquez, Gearbox fault diagnosis based on deep random forest fusion of acoustic and vibratory signals, Mech. Syst. Signal Processing 76 (2016) 283–293.
[88] S.-Y. Shao, W.-J. Sun, R.-Q. Yan, P. Wang, R.X. Gao, A deep learning approach for fault diagnosis of induction motors in manufacturing, Chinese J. Mech. Eng. 30 (6) (2017) 1347–1356.
[89] L. Zhang, H. Gao, J. Wen, S. Li, Q. Liu, A deep learning-based recognition method for degradation monitoring of ball screw with multi-sensor data fusion, Microelectron. Reliab. 75 (2017) 215–222.
[90] X. Wang, J. Huang, G. Ren, D. Wang, A hydraulic fault diagnosis method based on sliding-window spectrum feature and deep belief network, J. Vibroengineering 19 (6) (2017) 4272–4284.
[91] D. Han, N. Zhao, P. Shi, A new fault diagnosis method based on deep belief network and support vector machine with Teager-Kaiser energy operator for bearings, Adv. Mech. Eng. 9 (12) (2017), 1687814017743113.
[92] P. Wang, R.X. Gao, R. Yan, A deep learning-based approach to material removal rate prediction in polishing, CIRP Ann.-Manuf. Technol. 66 (1) (2017) 429–432.
[93] Z. Chen, S. Deng, X. Chen, C. Li, R.-V. Sanchez, H. Qin, Deep neural networks-based rolling bearing fault diagnosis, Microelectron. Reliab. 75 (2017) 327–333.
[94] Z. Gao, C. Ma, D. Song, Y. Liu, Deep quantum inspired neural network with application to aircraft fuel system fault diagnosis, Neurocomputing 238 (2017) 13–23.
[95] H. Oh, J.H. Jung, B.C. Jeon, B.D. Youn, Scalable and unsupervised feature engineering using vibration-imaging and deep learning for rotor system diagnosis, IEEE Trans. Industr. Electron. 65 (4) (2018) 3539–3549.
[96] M. Ma, X. Chen, S. Wang, Y. Liu, W. Li, Bearing degradation assessment based on weibull distribution and deep belief network, in: Proceedings of 2016 International Symposium of Flexible Automation (ISFA), 1–4, 2016.
[97] S. Shao, W. Sun, P. Wang, R.X. Gao, R. Yan, Learning features from vibration signals for induction motor fault diagnosis, in: Proceedings of 2016 International Symposium of Flexible Automation (ISFA), 1–6, 2016.
[98] Y. Fu, Y. Zhang, H. Qiao, D. Li, H. Zhou, J. Leopold, Analysis of feature extracting ability for cutting state monitoring using deep belief networks, Procedia CIRP 31 (2015) 29–34.
[99] P. Tamilselvan, P. Wang, Failure diagnosis using deep belief learning based health state classification, Reliab. Eng. Syst. Saf. 115 (2013) 124–135.
[100] P. Tamilselvan, Y. Wang, P. Wang, Deep belief network based state classification for structural health diagnosis, in: 2012 IEEE Aerospace Conference, 1–11, 2012.

[101] J. Tao, Y. Liu, D. Yang, Bearing fault diagnosis based on deep belief network and multisensor information fusion, Shock Vib. (2016) 9, Article ID 9306205.
[102] Z. Chen, C. Li, R.-V. Sánchez, Multi-layer neural network with deep belief network for gearbox fault diagnosis, J. Vibroeng. 17 (5) (2015) 2379–2392.
[103] M. Gan, C. Wang, et al, Construction of hierarchical diagnosis network based on deep learning and its application in the fault pattern recognition of rolling element bearings, Mech. Syst. Signal Processing 72 (2016) 92–104.
[104] H. Oh, B.C. Jeon, J.H. Jung, B.D. Youn, Unsupervised feature extraction scheme by deep learning, in: Annual Conference of the Prognostic and Health Management Society, PHM Society, 1–8, 2016.
[105] C. Zhang, J.H. Sun, K.C. Tan, Deep belief networks ensemble with multi-objective optimization for failure diagnosis, in: Systems, Man, and Cybernetics (SMC), 2015 IEEE International Conference on, IEEE, 32–37, 2015.
[106] C. Zhang, P. Lim, A. Qin, K.C. Tan, Multiobjective deep belief networks ensemble for remaining useful life estimation in prognostics, IEEE Trans. Neural Networks Learn. Syst. 28 (10) (2017) 2306–2318.
[107] R. Liu, G. Meng, B. Yang, C. Sun, X. Chen, Dislocated time series convolutional neural architecture: an intelligent fault diagnosis approach for electric machine, IEEE Trans. Industr. Inf. 13 (2) (2017) 1310–1320.
[108] P. Wang, Ananya, R. Yan, R.X. Gao, Virtualization and deep recognition for System Fault Classification, J. Manuf. Syst. 44 (2017) 310–316.
[109] O. Janssens, V. Slavkovikj, B. Vervisch, K. Stockman, M. Loccufier, S. Verstockt, R. Van de Walle, S. Van Hoecke, Convolutional neural network based fault detection for rotating machinery, J. Sound Vib. 377 (2016) 331–345.
[110] C. Lu, Z. Wang, B. Zhou, Intelligent fault diagnosis of rolling bearing using hierarchical convolutional network based health state classification, Adv. Eng. Inform. 32 (2017) 139–151.
[111] G.S. Babu, P. Zhao, X.-L. Li, Deep convolutional neural network based regression approach for estimation of remaining useful life, in: International Conference on Database Systems for Advanced Applications, Springer, 214–228, 2016.
[112] X. Ding, Q. He, Energy-fluctuated multiscale feature learning with deep convnet for intelligent spindle bearing fault diagnosis, IEEE Trans. Instrum. Meas. 66 (8) (2017) 1926–1935.
[113] X. Guo, L. Chen, C. Shen, Hierarchical adaptive deep convolution neural network and its application to bearing fault diagnosis, Measurement 93 (2016) 490–502.
[114] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324.
[115] W. Sun, B. Yao, N. Zeng, B. Chen, Y. He, X. Cao, W. He, An intelligent gear fault diagnosis methodology using a complex wavelet enhanced convolutional neural network, Materials 10 (7) (2017) 790.
[116] J. Wang, j. Zhuang, L. Duan, W. Cheng, A multi-scale convolution neural network for featureless fault diagnosis, in: Proceedings of 2016 International Symposium of Flexible Automation (ISFA), 1–6, 2016c.
[117] Z. Chen, C. Li, R.-V. Sanchez, Gearbox fault identification and classification with convolutional neural networks, Shock Vib (2015) 10, Atricle ID 390134.
[118] D. Weimer, B. Scholz-Reiter, M. Shpitalni, Design of deep convolutional neural network architectures for automated feature extraction in industrial inspection, CIRP Annals-Manufacturing Technology.
[119] H.-Y. Dong, L.-X. Yang, H.-W. Li, Small fault diagnosis of front-end speed controlled wind generator based on deep learning, WSEAS Trans. Circuits Syst. 15 (2016) 64–72.
[120] W. You, C. Shen, X. Guo, X. Jiang, J. Shi, Z. Zhu, A hybrid technique based on convolutional neural network and support vector regression for intelligent diagnosis of rotating machinery, Adv. Mech. Eng. 9 (6) (2017) 168–176.
[121] K.B. Lee, S. Cheon, C.O. Kim, A convolutional neural network for fault classification and diagnosis in semiconductor manufacturingprocesses, IEEE Trans. Semicond. Manuf. 30 (2) (2017) 135–142.
[122] L. Wen, X. Li, L. Gao, Y. Zhang, A new convolutional neural network based data-driven fault diagnosis method, IEEE Transactions on Industrial Electronics doi10.1109/TIE.2017.2774777.
[123] S. Li, G. Liu, X. Tang, J. Lu, J. Hu, An ensemble deep convolutional neural network model with improved DS evidence fusion for bearing fault diagnosis, Sensors 17 (8) (2017) 1729.
[124] S.K. Singh, S. Kumar, J. Dwivedi, Compound fault prediction of rolling bearing using multimedia data, Multimedia Tools Appl. (2017) 1–18.
[125] D. Verstraete, A. Ferrada, E.L. Droguett, V. Meruane, M. Modarres, Deep learning enabled fault diagnosis using time-frequency image analysis of rolling element bearings, Shock and Vibration (2017) 17, AtricleID 506765.
[126] T. Ince, S. Kiranyaz, L. Eren, M. Askar, M. Gabbouj, Real-time motor fault detection by 1-D convolutional neural networks, IEEE Trans. Industr. Electron. 63 (11) (2016) 7067–7075.
[127] O. Abdeljaber, O. Avci, S. Kiranyaz, M. Gabbouj, D.J. Inman, Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks, J. Sound Vib. 388 (2017) 154–170.
[128] L. Jing, M. Zhao, P. Li, X. Xu, A convolutional neural network based feature learning and fault diagnosis method for the condition monitoring of gearbox, Measurement 111 (2017) 1–10.
[129] W. Zhang, C. Li, G. Peng, Y. Chen, Z. Zhang, A deep convolutional neural network with new training methods for bearing fault diagnosis under noisy environment and different working load, Mech. Syst. Signal Processing 100 (2018) 439–453.
[130] W. Zhang, G. Peng, C. Li, Y. Chen, Z. Zhang, A new deep learning model for fault diagnosis with good anti-noise and domain adaptation ability on raw vibration signals, Sensors 17 (2) (2017) 425.
[131] Y. Li, N. Wang, J. Shi, J. Liu, X. Hou, Revisiting batch normalization for practical domain adaptation, arXiv preprint arXiv:1603.04779.
[132] W. Sun, R. Zhao, R. Yan, S. Shao, X. Chen, Convolutional discriminative feature learning for induction motor fault diagnosis, IEEE Trans. Industr. Inf. 13 (3) (2017) 1350–1359.
[133] D. Cabrera, F. Sancho, C. Li, M. Cerrada, R.-V. Sánchez, F. Pacheco, J.V. de Oliveira, Automatic feature extraction of time-series applied to fault severity assessment of helical gearbox in stationary and non-stationary speed operation, Appl. Soft Comput. 58 (2017) 53–64.
[134] H. Shao, H. Jiang, H. Zhang, T. Liang, Electric locomotive bearing fault diagnosis using novel convolutional deep belief network, IEEE Trans. Industr. Electron. 65 (5) (2018) 4290–4300.
[135] M. Zhao, M. Kang, B. Tang, M. Pecht, Deep residual networks with dynamically weighted wavelet coefficients for fault diagnosis of planetary gearboxes, IEEE Trans. Ind. Electron. (2017).
[136] J. Pan, Y. Zi, J. Chen, Z. Zhou, B. Wang, LiftingNet: a novel deep learning network with layerwise feature learning from noisy mechanical data for fault classification, IEEE Trans. Ind. Electron. (2017), https://doi.org/10.1109/TIE.2017.2767540.
[137] M. Yuan, Y. Wu, L. Lin, Fault diagnosis and remaining useful life estimation of aero engine using LSTM neural network, in: 2016 IEEE International Conference on Aircraft Utility Systems (AUS), 135–140, 2016.
[138] R. Zhao, J. Wang, R. Yan, K. Mao, Machine health monitoring with LSTM networks, in: IEEE International Conference on Sensing Technology, 1–6, 2016.
[139] R. Zhao, R. Yan, J. Wang, K. Mao, Learning to monitor machine health with convolutional bi-directional lstm networks, Sensors 17 (2) (2017) 273.
[140] R. Zhao, D. Wang, R. Yan, K. Mao, F. Shen, J. Wang, Machine health monitoring using local feature-based gated recurrent unit networks, IEEE Trans. Industr. Electron. 65 (2) (2017) 1539–1548.
[141] P. Malhotra, A. Ramakrishnan, G. Anand, L. Vig, P. Agarwal, G. Shroff, Multi-sensor prognostics using an unsupervised health index based on LSTM Encoder-Decoder, arXiv preprint arXiv:1608.06154.
[142] V. Stojanovic, N. Nedic, Robust identification of OE model with constrained output using optimal input design, J. Franklin Inst. 353 (2) (2016) 576–593.
[143] V. Stojanovic, V. Filipovic, Adaptive input design for identification of output error model with constrained output, Circuits, Systems, Signal Processing 33 (1) (2014) 97–113.

[144] V. Filipovic, N. Nedic, V. Stojanovic, Robust identification of pneumatic servo actuators in the real situationsRobuste Identifikation von pneumatischen Servo-Aktuatoren in der realen Situationen, Forsch. Ingenieurwes. 75 (4) (2011) 183–196.
[145] phm society, 2010 PHM Data Challenge, URLhttps://www.phmsociety.org/competition/phm/10, 2010
[146] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, arXiv preprint arXiv:1512.03385.
[147] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L.F.-F., ImageNet: A Large-Scale Hierarchical Image Database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, 248–255, 2009.
[148] L.v.d. Maaten, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (2008) 2579–2605.
[149] J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, H. Lipson, Understanding neural networks through deep visualization, arXiv preprint arXiv:1506.06579.
[150] F. Shen, C. Chen, R. Yan, R.X. Gao, Bearing fault diagnosis based on SVD feature extraction and transfer learning classification, in: 2015 IEEE Prognostics and System Health Management Conference (PHM), 1–6, 2015.
[151] J. Xie, L. Zhang, L. Duan, J. Wang, On cross-domain feature fusion in gearbox fault diagnosis under various operating conditions based on Transfer Component Analysis, in: IEEE International Conference on Prognostics and Health Management (ICPHM), 1–6, 2016.
[152] W. Lu, B. Liang, Y. Cheng, D. Meng, J. Yang, T. Zhang, Deep model based domain adaptation for fault diagnosis, IEEE Trans. Industr. Electron. 64 (3) (2017) 2296–2305.
[153] W. Mao, L. He, Y. Yan, J. Wang, Online sequential prediction of bearings imbalanced fault diagnosis by extreme learning machine, Mech. Syst. Signal Processing 83 (2017) 450–473.
[154] L. Duan, M. Xie, T. Bai, J. Wang, A new support vector data description method for machinery fault diagnosis with unbalanced datasets, Expert Syst. Appl. 64 (2016) 239–246.
[155] C. Huang, Y. Li, C. Change Loy, X. Tang, Learning deep representation for imbalanced classification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 5375–5384, 2016.
[156] Y. Yan, M. Chen, M.-L. Shyu, S.-C. Chen, Deep Learning for Imbalanced Multimedia Data Classification, in: 2015 IEEE International Symposium on Multimedia (ISM), IEEE, 483–488, 2015.