1684. Multi-layer neural network with deep belief network for gearbox fault diagnosis

Zhiqiang Chen[1], Chuan Li[2], René-Vinicio Sánchez[3]
[1, 2]Chongqing Engineering Laboratory for Detection Control and Integrated System,
Chongqing Technology and Business University, Chongqing, China
[2]Research Center of System Healthy Maintenance,
Chongqing Technology and Business University, Chongqing, China
[3]Department of Mechanical Engineering, Universidad Politécnica Salesiana, Cuenca, Ecuador
[2]Corresponding author
E-mail: [1]*wilber_chen@hotmail.com*, [2]*chuanli@21cn.com*, [3]*rsanchezl@ups.edu.ec*

Abstract. Identifying gearbox damage categories, especially for early faults and combined faults, is a challenging task in gearbox fault diagnosis. This paper presents multiple classifiers based on multi-layer neural networks (MLNN) to implement vibration signals for fault diagnosis in gearbox. A MLNN-based learning architecture using deep belief network ($MLNN_{DBN}$) is proposed for gearbox fault diagnosis. Training process of the proposed learning architecture includes two stages: A deep belief network is constructed firstly, and then is trained; after a certain amount of epochs, the weights of deep belief network are used to initialize the weights of the constructed MLNN; at last, the trained MLNN is used as classifiers to classify gearbox faults. Multidimensional feature sets including time-domain, frequency-domain features are extracted to reveal gear health conditions. Experiments with different combined faults were conducted, and the vibration signals were captured under different loads and motor speeds. To confirm the superiority of $MLNN_{DBN}$ in fault classification, its performance is compared with other MLNN-based methods with different fine-tuning schemes and relevant vector machine. The achieved accuracy indicates that the proposed approach is highly reliable and applicable in fault diagnosis of industrial reciprocating machinery.

Keywords: neural network, deep belief network, gearbox, vibratory signal.

1. Introduction

Gearboxes play crucial roles in the mechanical transmission systems, are used to transmit power between shafts and are expected to work 24 hours a day in the production system. Any failures with the gearboxes may introduce unwanted downtime, expensive repair and even human casualties. Therefore it is essential to detect and diagnose faults in the initial stage [1-4]. As an effective component for the condition-based maintenance, the fault diagnosis has gained much attention for the safe operations of the gearboxes. The gearbox conditions can be reflected by such measurements as vibratory, acoustic, thermal, electrical and oil-based signals [5-8]. Among above symptoms, the vibration analysis is the most commonly-used technique for the reason that every machine is considered to have a normal spectrum until there is a fault, where the spectrum changes [9, 10]. The vibration signals have been proven effective to reflect the healthy condition of the rotating machinery.

Various studies exist, of vibrations-based algorithms for detection and diagnostics of faults in gearboxes, among these are support vector machines and artificial neural network. A support vector machines based envelope spectrum was proposed by Guo et al. [11] to classify three health conditions of the planetary gearboxes. An intelligent diagnosis model based on wavelet support vector machine (SVM) and immune genetic algorithm (IGA) was proposed for the gearbox fault diagnosis [12]. The IGA was developed to determine the optimal parameters for the wavelet SVM with the highest accuracy and generalization ability. Tayarani-Bathaie et al. [13] suggested a dynamic neural network to diagnose the gas turbine fault. The artificial neural network combining with empirical mode decomposition was applied for automatic bearing fault diagnosis based on

vibration signals [14]. Among all the typical classifiers, the support vector classification (SVC) family (i.e., the standard SVC and its variants) attracted much attention due to their extraordinary classification performance. According to the researches, the SVM family received good results in comparison with the peer classifiers.

Multi-layer neural networks, typically used in supervised learning to make a prediction or classification. As Y. Bengio in the literature [15, 16] reported, the gradient-based training of supervised multi-layer neural networks (starting from random initialization) gets easily stuck in "apparent local minima or plateaus". That is to restrict its application for gearbox fault diagnosis. Since 2006, deep networks have been applied with success in classification tasks and other fields such as in regression, dimensionality reduction, and modeling textures [17], but few were used for the fault diagnosis cases. Tran et al. [18] introduced the application of the deep belief networks to diagnose reciprocating compressor valves. Tamilselvan and Wang [19] employed the deep belief learning based health state classification for iris dataset, wine dataset, Wisconsin breast cancer diagnosis dataset, and Escherichia coli dataset. The limited reports show the deep learning structure for the fault diagnosis with commonly one modality feature.

In this work, we combine deep learning strategies with multi-layer neural network for the identification and classification of gearbox faults, where complicated faults are considered that a fault signal usually includes one or several basic faults. The typical multi-layer neural network (MLNN) [20] is used firstly for the identification and classification of gearbox faults, where batch training strategy is used to train the neural network. Then several fine-tuning schemes for improving the typical multi-layer neural network by preventing co-adaptation of feature detectors are suggested for the gearbox faults identification and classification. At last, a MLNN-based deep learning technique is proposed, where the weights of neural network are initialized using deep belief network (DBN) [21].

To validate the robustness of the MLNN-based approaches, a fault condition pattern library is constructed, which has 58 kinds of combined fault patterns. 20 test cases are used and each test case has 12 kinds of condition pattern that are randomly selected from the pattern library. A large number of experiments show the multi-layer neural network with DBN (MLNN$_{DBN}$) has most excellent performance and achieves to avoid falling into "apparent local minima or plateaus".

The rest of this paper is structured as follows. Section 2 introduces the used methodologies including multi-layer neural networks and proposed MLNN-based learning architecture using deep belief network; Section 3 introduces feature representations of vibratory signals. Section 4 presents the implementation of the classifier based on multi-layer neural network; Section 5 introduces experimental setup; results and discussion are presented in Section 6. Finally some conclusions are drawn.

2. Methodologies

In this section, multi-layer neural network is firstly introduced, and then several fine-tuning schemes are explained. At last, MLNN-based learning architecture using deep belief network (MLNN$_{DBN}$) is proposed for gearbox fault diagnosis.

2.1. Multi-layer neural network

Multi-layer neural network is typically used in supervised learning to make a prediction or classification, through a series of layers, each of which combines an affine operation and a non-linearity. A typical set of equations for multi-layer neural network [17, 20] is the following. As illustrated in Fig. 1, layer k computes an output vector $\mathbf{h}^k$ using the output $\mathbf{h}^{k-1}$ of the previous layer, starting with the input $\mathbf{x} = h^0$:

$$\mathbf{h}^k = \tanh(\mathbf{b}^k + \boldsymbol{W}^k \mathbf{h}^{k-1}), \tag{1}$$

with parameters $\mathbf{b}^k$ (a vector of offsets) and $\mathbf{W}^k$ (a matrix of weights). The $\tanh(\cdot)$ is can be replaced by $\text{sigm}(u) = 1/(1 + e^{-u}) = (\tanh(u) + 1)/2$ or other saturating non-linearities [17]. The top layer output $\mathbf{h}^l$ is used for making a prediction and is combined with a supervised target y into a loss function $L(\mathbf{h}^l, y)$, typically convex in $\mathbf{b}^l + W^l\mathbf{h}^{l-1}$. The output layer might have a non-linearity different from the one used in other layers, e.g., the softmax:

$$\mathbf{h}_i^l = \frac{e^{b_i^l + W_i^l\mathbf{h}^{l-1}}}{\sum_j e^{b_j^l + W_j^l\mathbf{h}^{l-1}}}, \tag{2}$$

where W_i^l is the ith row of W^l, $\mathbf{h}_i^l$ is positive and $\sum_i \mathbf{h}_i^l = 1$. The softmax output $\mathbf{h}_i^l$ can be used as estimator of $P(Y = i|x)$, with the interpretation that Y is the class associated with input pattern $\mathbf{x}$. In this case one often uses the negative conditional log-likelihood $L(\mathbf{h}^l, y) = -\log P(Y = y|\mathbf{x}) = -\log \mathbf{h}_y^l$ as a loss, whose expected value over $(\mathbf{x}, y)$ pair is to be minimized [17]. The training of a MLNN is usually accomplished by using back propagation (BP) algorithm that involves two phases [20]:

– Forward phase: During this phase the free parameters of the network are fixed, and the input signal is propagated through the network of Fig. 1 layer by layer. The forward phase finishes with the computation of an error signal:

$$e_i = d_i - y_i, \tag{3}$$

where d_i is the desired response and y_i is the actual output produced by the network in response to the input $\mathbf{x}_i$.

– Backward phase: During this second phase, the error signal e_i is propagated through the network of Fig. 1 in the backward direction. It is during this phase that adjustments are applied to the free parameters of the network so as to minimize the error e_i in a statistical sense.
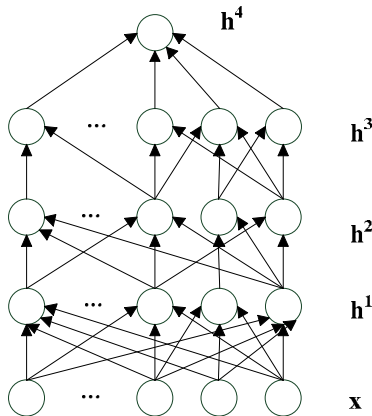

Fig. 1. Multi-layer neural network

2.2. Existing fine-tuning schemes for typical MLNN

As Y. Bengio in the literature [15, 16] reported, the gradient-based training of supervised multi-layer feed-forward neural network gets easily stuck in "apparent local minima or plateaus" (starting from random initialization). To improve the global exploration ability of MLNN, and avoid falling into the local minima, G. E. Hinton et al. proposed overfitting can be reduced to improve neural network by preventing co-adaptation of feature detectors [22]. The following several fine-tuning schemes try to improve the performance of the typical MLNN:

Scheme 1: During stochastic gradient descent procedure for training the networks on mini-batches of training cases, a penalty term is added that is normally used to prevent the weights from growing too large. The deltas of updating weights multiply by a penalty factor. Finally, the delta rule for updating a weight assigned to a given neuron is as follows:

$$\Delta W_i = \eta (\Delta W_i + p W_i), \tag{4}$$

where η is a learning rate, p is the penalty factor. They are set to 2 and 1×10^{-4}, respectively in this case.

Scheme 2: On each presentation of each training case, each hidden unit is randomly omitted from the network with a probability of 0.5, which is called "dropout" by G. E. Hinton [22].

Scheme 3: Activation function of hidden layers uses sigmoid function. Learning rate note: typically needs to be lower when using "sigm" activation function and non-normalized inputs. So learning rate is set to 1 for this case.

Scheme 4: output unit use "softmax" function.

In this work, we also evaluate the above fine-tuning schemes for gearbox fault diagnosis.

2.3. MLNN-based learning architecture using deep belief network (MLNN$_{DBN}$)

In this subsection, we propose a new fine-tuning strategy to improve the performance of typical MLNN. Training process of the proposed learning architecture includes two stages: A deep belief network is constructed firstly, and then is trained; after a certain amount of epochs, the weights of deep belief network are used to initialize the weights of the constructed MLNN; at last, the trained MLNN is used as classifiers to classify gearbox faults. The outline of MLNN-based learning architecture using deep belief network is as Fig. 2.
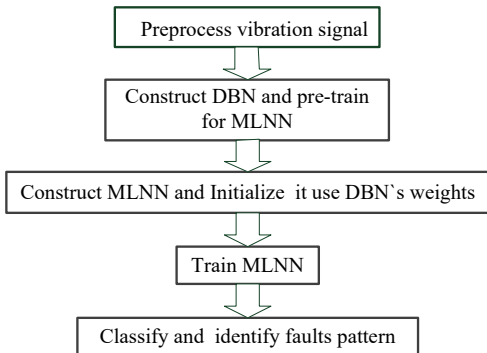

Fig. 2. Outline of MLNN-based learning architecture using deep belief network
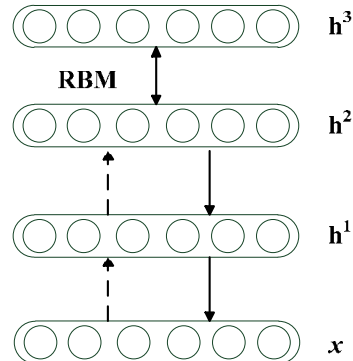
Fig. 3. Deep belief network

Deep belief networks (DBNs) [23] can be viewed as a composition of simple learning modules each of which is a restricted type of Boltzmann machine that contains a layer of visible units that represent the data and a layer of hidden units that learn to represent features that capture higher-order correlations in the data. It is a model in which the top two hidden layers form an undirected associative memory (see Fig. 3) and the remaining hidden layers form a directed acyclic graph that converts the representations in the associative memory into observable variables such as the pixels of an image. The two layers are connected by a matrix of symmetrically weighted connections, $\mathbf{W}$, and there are no connections within a layer. Given a vector of activities v for the visible units, the hidden units are all conditionally independent so it is easy to sample a vector, h, from the factorial posterior distribution over hidden vectors, $p(\mathbf{h}|\mathbf{v}, \mathbf{W})$. It is also easy to sample from $p(\mathbf{v}|\mathbf{h}, \mathbf{W})$. By starting with an observed data vector on the visible units and alternating several times between sampling from $p(\mathbf{h}|\mathbf{v}, \mathbf{W})$ and $p(\mathbf{v}|\mathbf{h}, \mathbf{W})$, it is easy to get a

learning signal. This signal is simply the difference between the pairwise correlations of the visible and hidden units at the beginning and end of the sampling.

The key idea of deep belief nets is that the weights, **W**, learned by a restricted Boltzmann machine define both $p(v|h, W)$, and the prior distribution over hidden vectors, $p(h, W)$, so the probability of generating a visible vector, v, can be written as:

$$p(v) = \sum_h p(h|W)p(v|h, W).$$

(5)

The training algorithm for DBNs proceeds as follows. Let **X** be a matrix of inputs, regarded as a set of feature vectors.

1) Train a restricted Boltzmann machine (RBM, see its details in the literature [24]) on **X** to obtain its weight matrix, **W**. Use this as the weight matrix for between the lower two layers of the network.

2) Transform **X** by the RBM to produce new data **X′**.

3) Repeat this procedure with **X** ← **X′** for the next pair of layers, until the top two layers of the network are reached.

3. Feature representations of vibratory signals

The gearbox condition can be reflected by the information included in different frequency and time domain. The features in frequency and time domain are obtained from the set of signals obtained from the measurements of the vibrations at different speeds and loads, which can be used as input parameters for the multi-layer neural network.

3.1. Time-domain feature extraction

The time-domain signal collected from a gearbox usually changes when damage occurs in a gear or bear. Both its amplitude and distribution may be different from those of the time-domain signal of a normal gear or bear. Root mean square reflects the vibration amplitude and energy in time domain. Standard deviation, skewness and kurtosis may be used to represent the time series distribution of the signal in the time domain.

First, four time-domain features, namely, standard deviation, mean value, skewness and kurtosis are calculated. They are defined as follows.

1) Mean value:

$$\bar{x} = \frac{1}{N} \sum_{n=1}^{N} x(n).$$

(6)

2) Standard deviation:

$$\sigma = \sqrt{\frac{1}{N} \sum_{n=1}^{N} (x(n) - \bar{x})^2}.$$

(7)

3) Skewness:

$$S = \frac{E(x - \bar{x})^3}{\sigma^3}.$$

(8)

4) Kurtosis:

$$K = \frac{1}{N} \sum_{n=1}^{N} \frac{(x(n) - \bar{x})^4}{\sigma^4}. \tag{9}$$

3.2. Frequency-domain feature extraction

The time domain signal was multiplied by a Hanning window to obtain the FFT spectrum. The spectrum was divided in multiple bands, and the root mean square value (RMS) was calculated for each. The RMS value of each band is used as feature representation in the spectrum domain. With the objective of reducing the amount of input data the spectrum was split in multiple bands, because with this number of bands the root mean square (RMS) values keep track of the energy in the spectrum peaks [25]. RMS value is evaluated with Ep. (10), where N is the number of samples of each frequency band:

$$FFT_{rms} = \sum_{n=1}^{N} FFT(n). \tag{10}$$

Vectors of the features of the preprocessed signal are formed as follows: N_{RMS} RMS values, standard deviation, skewness, kurtosis, rotation frequency and applied load measurements, which are used as input parameters for neural networks. The outline of feature representation of vibratory signals is as Fig. 4. In this work, the frequency range is 0 to 22050 Hz and the size of the data vector in the frequency is 18000 samples. The spectrum is divided in N_{RMS} frequency bands, $N_{RMS} = 251$.
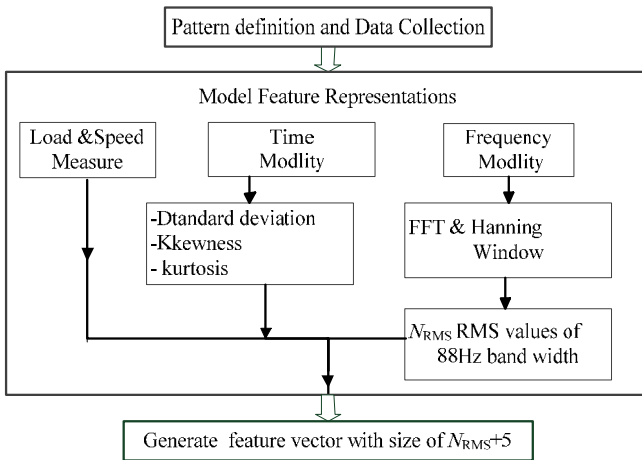

Fig. 4. Feature representation

4. Implemented classifier based on multi-layer neural network

4.1. Typical MLNN-based classifier

To validate the performance of the proposed MLNN$_{DBN}$-based classifier for gearbox fault diagnosis, we firstly implement the classifier based on typical MLNN. A batch training strategy is used to train the neural network, where the weights of nets are shared by a batch of training samples with mini batches of size. In other word, after training the neural network with a batch of samples, its errors, weights and biases are updated.

After performing a series of parameter tunings, a net of typical MLNN with 3 layers is constructed as follows: 256 inputs, 12 outputs, 1 hidden layer of 80 neurons, 1 output layer of 12

neurons; all neurons use the tan-sigmoid as the transfer function. The learning rate and momentum of the network are set to 2, 0.5, respectively. Activation function of hidden layers is tanh function, and that of output unit is sigmoid function. The size of batch training is 100. The number of samples is 12000. The number of iteration of training is 85. The training of the network is accomplished by using back propagation (BP) algorithm. The pseudo-code of the MLNN-based classifier is as Fig. 5.

```
MLNN-based Algorithm:
    Preprocess vibratory signal as Fig. 4
        -Number of Samples: 12000, 60% for Training and 40% for Test;
        -Training Set: train_x;
        -Label of Training Data: train_y;
    Construct net of MLNN
        -256 inputs, 1 hidden layer of 85 neurons
        -1 output layer of 12 neurons
    Train Neural Network:
        For i = 1 : numepochs
            kk = randperm(m);
            For l = 1 : numbatches
                batch_x = train_x( kk((l - 1) * batchsize + 1 : l * batchsize), :);
                batch_y = train_y( kk((l - 1) * batchsize + 1 : l * batchsize), :);
                % performs a feedforward pass
                %returns a net structure with updated
                nn = nnff(nn, batch_x, batch_y);
                %performs backpropagation pass
                %returns an net structure with updated delta of weights
                nn = nnbp(nn);
                %updates weights and biases with calculated gradients
                % returns a net structure with updated weights and biases
                nn = nnapplygrads(nn);
            End
        End
    Evaluate and Test Neural Network
    Classify Signals and Identify Faults
```

Fig. 5. Pseudo-code of MLNN-based classifier

4.2. MLNN$_{DBN}$-based classifier

As above described, DBN is employed to initialize the weight of MLNN. After pre-training to use DBN, the weights of MLNN are initialized to use the weights of DBN. The pseudo-code of the MLNN$_{DBN}$-based classifier is indicated in Fig. 6. After performing a series of parameter tunings, the architecture of DBN is defined as follows: Input layer: 256 neurons; two hidden layers, each one has 30 neurons; the size of batch training and learning rate are set to 100 and 1, respectively. After a large number of trials, it is suitable that the number of iteration of pre-training is 1.

The architecture of MLNN used here is defined as follows: input layer: 256 neurons; two hidden layers, each one has 30 neurons; 1 output layer of 12 neurons; all neurons of the neural network use the sigmoid as the transfer function; the learning rate, the size of batch training and the number of iteration of training are set to 1, 100, and 85, respectively.

5. Experimental setup

In this subsection, we firstly explain the gearbox fault diagnosis experimental platform. Fig. 7 indicates the internal configuration of the gearbox and positions for accelerometers. There are 3 shafts and 4 gears composing a two-stage transmission of the gearbox. An input gear ($Z_1 = 27$,

modulus = 2, Φ of pressure = 20) was installed on the input shaft. Two intermediate gears ($Z_2 = Z_3 = 53$) were installed on an intermediate shaft for the transmission for the transmission between the input gear and the output gear ($Z_4 = 80$, installed on the output shaft). The faulty components used in the experiments included gears Z_1, Z_2, Z_3, Z_4, bearing B_1, B_2, B_3, B_4 as labeled in Fig. 7(a). Test's conditions are described in the Table 1. The vibration signal is obtained from the measurements of a vertically allocated accelerometer in the gearbox case.

```
MLNN_DBN Algorithm:
    Preprocess vibratory signal as Fig. 4
        -Number of Samples: 12000, 60% for Training and 40% for Test;
        -Training Set: train_x;
        -Label of Training Data: train_y;
    Setup of DBN Net
        -256 inputs, 2 hidden layer of 30 neurons
        -Size of Batch:100; Learning Rate: 1
    Setup of MLNN Net
        -256 inputs, 2 hidden layer of 30 neurons
        -1 output layer of 12 neurons
        -Active function of hidden layer: 'sigmoid'; Learning Rate: 1
    Train DBN Net:
        %Train the first layer as an RBM model
        dbn.rbm{1} = rbmtrain(dbn.rbm{1}, train_x);
        %Train the other layer as an RBM model
        x = train_x;
        N = Number of layers of DBN Net
        for i = 2 : N
            %Use that first layer to obtain a representation of the input that
            %will be used as data for the second layer.
            x = rbmup(dbn.rbm{i - 1}, x)
            %Train the second layer as an RBM
            dbn.rbm{i} = rbmtrain(dbn.rbm{i}, x, opts);
        end
    Initialize weights of MLNN Net with that of DBN Net
    Train MLNN Net as Fig. 5
    Evaluate and Test Neural Network
    Classify Signals and Identify Faults
```

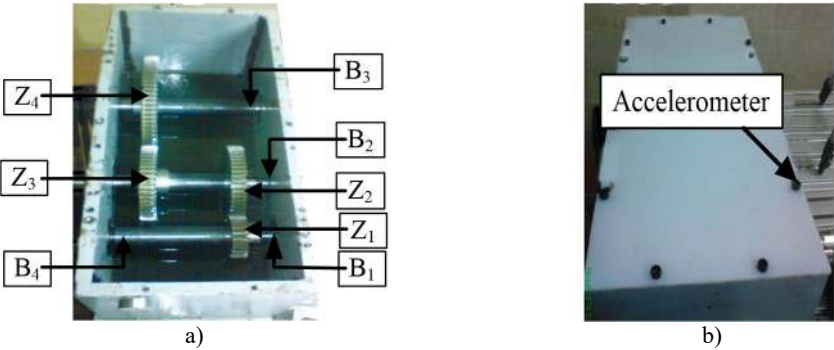**Fig. 6.** Pseudo-code of MLNN_DBN-based classifier

Fig. 7. a) The internal configuration of the gearbox; b) positions for accelerometers

Table 2 and 3 present the description of each fault condition of each component of the gearbox used in the experiment. We call them as basic condition pattern. In our experiment, a test case includes several basic condition patterns, which is a combination of multiple component faults. For example, the test case A shown in Table 4, includes following information of faults:

Gear Z_1: Gear with pitting on teeth;

Gear Z_2: Gear with face wear 0.5 [mm];

Bear B_3: Bearing with 4 pitting on outer ring;

Bear B_2: Bearing with 2 pitting on outer ring;

Gear Z_4 and Z_3, Bear B_1 and B_4: Normal.

Table 1. Test's conditions

Characteristic (C1)	Value
Sample frequency	44100 [Hz] (16 bits)
Sampled time	10 [s]
Power	1000 [W]
Minimum speed	700 [RPM]
Maximum speed	1600 [RPM]
Minimum load	250 [W]
Maximum load	750 [W]
Speeds	1760, 2120, 2480, 2840, 3200 [mm/s]
Loads	375, 500, 625, 750 [W]
Number of loads per test	10
Type of accelerometer	Uni-axial
Trademark	ACS
Model	ACS 3411LN
Sensibility	330 [mV/g]

To evaluate the performance of the proposed method for gearbox fault diagnosis, first, we constructed 12 condition patterns as listed in Table 4. Each pattern with 4 different load conditions and 5 different input speeds were applied during the experiments. For each pattern, load and speed condition, we repeated the tests for 5 times. In each time of the test, the vibratory signals were collected with 24 durations each of which covered 0.4096 sec.

Table 2. Nomenclature of gears fault

Designator	Description
1	Normal
2	Gear with face wear 0.4 [mm]
3	Gear with face wear 0.5 [mm]
4	Gear with chafing in tooth 50 %
5	Gear with chafing on tooth 100 %
6	Gear with pitting on tooth depth 0.05 [mm], width 0.5 [mm], large 0.05 [mm]
7	Gear with pitting on teeth
8	Gear with incipient fissure on 4 mm teeth to 25 % of profundity and angle of 45°
9	Gear teeth breakage 20 %
10	Gear teeth breakage 50 %
11	Gear teeth breakage 100 %

Table 3. Nomenclature of bears fault

Designator	Description
1	Normal
2	Bearing with 2 pitting on outer ring
3	Bearing with 4 pitting on outer ring
4	Bearing with 2 pitting on inner ring
5	Bearing with 4 pitting on inner ring
6	Bearing with race on inner ring
7	Bearing with 2 pitting on ball
8	Bearing with 2 pitting on ball

In addition, to further validate the performance of the proposed approaches, a fault condition pattern library was constructed, which has 58 kinds of condition pattern based on the basis patterns described in Table 2 and Table 3. Each condition pattern has more than one basic gearbox fault. 20 test cases are used to test the robustness of the all MLNN-based methods; each test case has 12 kinds of condition pattern that are randomly selected from the pattern library.

Table 4. Condition patterns of the experiment

No. patterns	Gear faults				Bear faults			
	Z_1	Z_2	Z_3	Z_4	B_1	B_2	B_3	B_4
A	7	3	1	1	1	2	3	1
B	7	3	6	8	1	1	1	1
C	5	5	1	1	6	7	2	1
D	7	1	1	1	6	7	2	1
E	1	2	1	1	1	6	3	1
F	1	3	1	1	1	5	3	1
G	2	9	1	1	6	7	3	1
H	5	5	1	1	6	3	2	4
I	2	6	1	1	6	5	2	1
J	1	11	1	1	1	3	4	1
K	1	1	1	1	1	6	3	1
L	1	1	1	1	1	1	3	1

6. Results and discussion

The training is done in first instance with the 12 patterns indicated in Table 4, using data with 12000 sample signals, where sixty percent of the samples set are used for the training of all MLNN-based approaches, and forty percent are used for testing. Table 5 shows their classification accuracy of the test case, including SVM classifier. We can see six MLNN-based approaches are effective for this test case, and are far better than SVM. The proposed $MLNN_{DBN}$ has 98.1 % classification accuracy and is superior to other MLNN-based approaches.

Table 5. Classification accuracy of test case indicated Table 4

Classification accuracy	$MLNN_{DBN}$	MLNN	$MLNN_{Scheme1}$	$MLNN_{Scheme2}$	$MLNN_{Scheme3}$	$MLNN_{Scheme4}$	SVM
	98.1 %	97.8 %	96.3 %	97.6 %	98.0 %	97.2 %	96.5 %

One test case cannot reflect the reliability and robustness of an algorithm. So to further validate the performance of the proposed approach, a fault condition pattern library was constructed. 20 test cases are used to test the robustness of the all MLNN-based methods, where each test case has 12 kinds of condition pattern that are randomly selected from the defined pattern library.

The experiment results of 20 test cases using above described seven learning techniques respectively are shown in Table 6. Among 20 test cases, the MLNN (typical multi-layer neural networks) has 6 test cases with bad classification accuracy, although it is effective for other 14 test cases whose classification accuracies are larger than 90 %. The classification accuracy of MLNN are even smaller than SVM for 4 test cases (#1, #4, #7 and #13). It is obviously that typical MLNN gets easily stuck in "apparent local minima or plateaus" in some cases and has not good robustness for gearbox faults diagnosis.

Four fine-tuning schemes presented in subsection 2.2 ($MLNN_{Scheme1}$, $MLNN_{Scheme2}$, $MLNN_{Scheme3}$ and $MLNN_{Scheme4}$) improve obviously the performance of the typical MLNN. Compared with the typical MLNN, $MLNN_{Scheme1}$ has 13 test cases, $MLNN_{Scheme2}$ has 14 test cases, and $MLNN_{Scheme3}$ has 15 test cases to be superior to it. $MLNN_{Scheme4}$ only improves that of the typical MLNN in two test cases (#4 and #12). However, the classification rates of each fine-tuning

scheme are smaller than 80 % for 5 test cases (#1, #4, #7, #13 and #20). They cannot still avoid falling into "apparent local minima or plateaus".

Classification accuracies of MLNN$_{DBN}$ for 20 test cases are presented in Table 6. The smallest one is 94.8 %, which is very excellent performance for gearbox fault diagnosis. Especially for the #1, #4, #7, #13 and #20 test case, the suggested MLNN$_{DBN}$ has true positive classification rate larger than 96 %, in which other MLNN-based methods cannot all effectively classify them. The mean, standard deviation, least, and most value of classification rate for 20 test cases are also indicated in Table 6. Experiment results in Table 6 show MLNN$_{DBN}$ achieves to avoid falling into "apparent local minima or plateaus".

Table 6. Classification accuracy of 20 test cases from the pattern library

No.	1	2	3	4	5	6	7	8
MLNN	54.5 %	99.0 %	99.0 %	62.9 %	99.0 %	98.8 %	67.4 %	98.9 %
MLNN$_{Scheme1}$	71.2 %	99.3 %	90.3 %	69.9 %	99.0 %	99.0 %	63.4 %	99.4 %
MLNN$_{Scheme2}$	67.3 %	99.4 %	98.6 %	68.7 %	99.2 %	96.7 %	68.0 %	96.4 %
MLNN$_{Scheme3}$	65.7 %	99.3 %	**99.2 %**	68.7 %	**99.3 %**	**99.3 %**	60.0 %	99.3 %
MLNN$_{Scheme4}$	47.9 %	97.7 %	96.3 %	75.9 %	96.8 %	97.3 %	59.7 %	96.5 %
SVM	73.5 %	96.1 %	96.8 %	95.5 %	98.0 %	73.2 %	96.1 %	96.8 %
MLNN$_{DBN}$	**98.1 %**	**99.1 %**	**99.0 %**	**99.0 %**	**99.1 %**	**98.8 %**	**98.6 %**	**99.0 %**
No.	9	10	11	12	13	14	15	16
MLNN	98.7 %	99.1 %	99.3 %	96.1 %	51.2 %	98.9 %	97.0 %	98.7 %
MLNN$_{Scheme1}$	96.9 %	38.9 %	99.3 %	99.1 %	54.4 %	99.2 %	96.0 %	99.5 %
MLNN$_{Scheme2}$	98.9 %	97.3 %	99.4 %	99.3 %	44.2 %	99.1 %	97.9 %	94.7 %
MLNN$_{Scheme3}$	98.8 %	90.1 %	99.0 %	99.2 %	53.1 %	98.8 %	98.1 %	99.0 %
MLNN$_{Scheme4}$	97.2 %	97.7 %	98.4 %	96.8 %	45.7 %	97.2 %	94.1 %	94.6 %
SVM	96.1 %	96.0 %	97.3 %	96.4 %	94.9 %	96.7 %	96.5 %	97.0 %
MLNN$_{DBN}$	**98.7 %**	**99.3 %**	**98.8 %**	**99.0 %**	**96.0 %**	**97.8 %**	**98.5 %**	**99.3 %**
No.	11	12	13	14	Mean	Std.	Least	Most
MLNN	98.1 %	89.1 %	93.6 %	71.8 %	88.6 %	16.7 %	51.2 %	99.3 %
MLNN$_{Scheme1}$	99.5 %	97.1 %	95.3 %	76.3 %	87.1 %	18.2 %	38.9 %	99.5 %
MLNN$_{Scheme2}$	99.2 %	91.5 %	95.6 %	72.4 %	89.2 %	15.9 %	44.2 %	99.4 %
MLNN$_{Scheme3}$	89.3 %	92.6 %	95.3 %	75.6 %	89.0 %	15.3 %	53.1 %	99.3 %
MLNN$_{Scheme4}$	96.3 %	89.1 %	89.9 %	71.2 %	86.8 %	15.2 %	45.7 %	98.4 %
SVM	95.1 %	96.3 %	93.3 %	96.4 %	94.9 %	6.09 %	73.2 %	98.0 %
MLNN$_{DBN}$	**98.3 %**	**94.8 %**	**95.6 %**	**98.7 %**	**98.3 %**	**1.3 %**	**94.8 %**	**98.7 %**

In addition, the MLNN$_{DBN}$ method was compared with "shallow" learning algorithms SVM. As for the SVM, one of the most important representatives in the "shallow" learning community, good classification results can be found for the gearbox fault diagnosis, which is similar with some existing researches [26]. The algorithm SVM is applied using the LibSVM [27]. The parameters for SVM are chosen as $C = 1$ and core (kernel) given by a radial basis $K(u, v) = \exp(-\gamma |u - v|^2)$ function where $\gamma = 0.5$. These parameters were found through a search, aiming at the best model for the SVM. Compared with other MLNN-based methods and SVM, we can see MLNN$_{DBN}$ has most excellent performance and is highly reliable.

Fig. 8 shows the convergence process of error rate on the test case indicated in Table 4 for different MLNN-based classifiers. After about 20 epochs of training, the error rate of each method is less than 0.1. However, it is obviously to exist "overfitting" while undergo more than 20 epochs of training except for MLNN$_{DBN}$. Especially MLNN$_{scheme2}$ has serious "overfitting" after 60 epochs of training. From Fig. 8, we can see that MLNN$_{DBN}$ is also robustness for gearbox fault diagnosis and overcome the problem of "overfitting".

Confusion matrix is an effective and visualization tool of the performance of a classification algorithm. Each column of the confusion matrix represents the instances in a predicted class (output class), while each row represents the instances in an actual class (target class). Fig. 9

presents the confusion matrix using MLNN$_{DBN}$ for 12 patterns indicated in Table 4. As shown in Fig. 9, it can be observed that the global percentage of true positive classification of the 12 condition patterns of faults is 98.1 % and the total error is 1.9 %. The smallest percentage of true positive classifications is obtained for type 3; this is because this kind of condition patterns with 6 basic faults. This is evident by observing the confusion matrix in which 34 times of type 4 are classified as type 3, noticing that mostly there is confusion between type 4 and type 3, in which they have 4 same basic faults. The percentage of true positive classification of Type 6, 7 and 12 are all 100 %. Confusion matrix in Fig. 9 shows the proposed MLNN$_{DBN}$ has very high percentage of true positive classification for the test case.
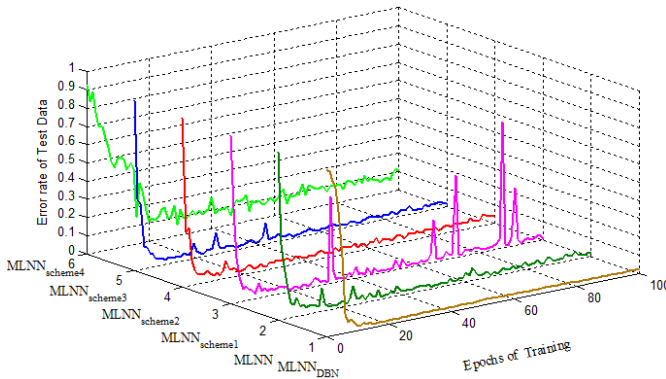
Fig. 8. The error rate on the test case indicated in Table 5 for different MLNN-based classifiers

Fig. 9. Confusion matrix using MLNN$_{DBN}$

All above experiments indicate MLNN-based learning technologies are efficient and effective for gearbox faults diagnosis. Among them, MLNN$_{DBN}$ has most effective performance and best robustness, which is highly reliable and applicable in fault diagnosis of industrial reciprocating machinery.

7. Conclusions

In this paper, multiple classifiers based on multi-layer neural network for the vibration measurements have been presented to diagnose the fault patterns of the gearbox. A MLNN-based learning architecture using deep belief network was proposed for gearbox fault diagnosis, which achieves to avoid falling into "apparent local minima or plateaus" and is highly reliable and

applicable in fault diagnosis of gearbox. Other MLNN-based learning technologies are efficient and effective for gearbox faults diagnosis, however they may be fall into "apparent local minima or plateaus" in some cases. This type of classifiers could make a contribution to maintenance routines for industrial systems, towards lowering costs and guarantying a continuous production system, and with the appropriate equipment, online diagnostics could be performed. With these methods, complex fault combination with different component fault can be classified. These ways the implemented classifier could be utilized with ease in other kinds of rotating machinery in general for the classification and diagnostics of mechanical faults from their initial stages and in an extended range of cases, using the vibration signals, allowing this way to have the most efficient control and planning of maintenance of rotating machines. For evaluating the proposed MLNN-based methods, the gearbox fault diagnosis experiments were also carried out using SVM technique. The results show that the proposed $MLNN_{DBN}$-based methods are superior to SVM technique.

Acknowledgements

References

[1] **Lei Y., Zuo M. J., He Z., Zi Y.** A multidimensional hybrid intelligent method for gear fault diagnosis. Expert Systems with Applications, Vol. 37, 2010, p. 1419-1430.

[2] **Yang Z., Hoi W. I., Zhong J.** Gearbox fault diagnosis based on artificial neural network and genetic algorithms. International Conference on System Science and Engineering (ICSSE), 2011, p. 37-42.

[3] **Aherwar P. Amit, Khalid MD. Saifullah** Vibration analysis techniques for gearbox diagnostic: a review. International Journal of Advanced Engineering Technology, Vol. 2008, 2008, p. 583982.

[4] **Li Chuan, Liang Ming, Wang Tianyang** Criterion fusion for spectral segmentation and its application to optimal demodulation of bearing vibration signals. Mechanical Systems and Signal Processing, 2015.

[5] **Wang D., Miao Q., Kang R.** Robust health evaluation of gearbox subject to tooth failure with wavelet decomposition. Journal of Sound and Vibration, Vol. 324, Issues 3-5, 2009, p. 1141-1157.

[6] **Toutountzakis T., Tan C. K., Mba D.** Application of acoustic emission to seeded gear fault detection. NDT&E International, Vol. 38, Issue 1, 2005, p. 27-36.

[7] **Ottewill J. R., Orkisz M.** Condition monitoring of gearboxes using synchronously averaged electric motor signals. Mechanical Systems and Signal Processing, Vol. 38, Issue 2, 2013, p. 482-498.

[8] **Younus A. M. D., Yang B. S.** Intelligent fault diagnosis of rotating machinery using infrared thermal image. Expert Systems with Applications, Vol. 39, Issue 2, 2012, p. 2082-2091.

[9] **Li C., Liang M.** Extraction of oil debris signature using integral enhanced empirical mode decomposition and correlated reconstruction. Measurement Science and Technology, Vol. 22, Issue 8, 2011, p. 085701.

[10] **Jayaswal P., Wadhwani A., Mulchandani K.** Machine fault signature analysis. International Journal of Rotating Machinery, Vol. 2008, 2008, p. 583982.

[11] **Guo L., Chen J., Li X.** Rolling bearing fault classification based on envelope spectrum and support vector machine. Journal of Vibration and Control, Vol. 15, Issue 9, 2009, p. 1349-1363.

[12] **Chen F., Tang B., Chen R.** A novel fault diagnosis model for gearbox based on wavelet support vector machine with immune genetic algorithm. Measurement, Vol. 46, Issue 1, 2013, p. 220-232.

[13] **Tayarani-Bathaie S. S., Vanini Z. N. S., Khorasani K.** Dynamic neural network-based fault diagnosis of gas turbine engines. Neurocomputing, Vol. 125, Issue 11, 2014, p. 153-165.

[14] **Leng B., Zhang X., Yao M., Xiong Z.** A 3D model recognition mechanism based on deep Boltzmann machines. Neurocomputing, Vol. 151, 2015, p. 593-602.

[15] **Bengio Y., Lamblin P., Popovici D., Larochelle H.** Greedy layer-wise training of deep networks. Advances in Neural Information Processing Systems, Vol. 19, 2007, p. 153-160.

[16] **Erhan D., Manzagol P.-A., Bengio Y., Bengio S., Vincent P.** The difficulty of training deep architectures and the effect of unsupervised pretraining. Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics, 2009, p. 153-160.

[17] **Bengio Yoshua** Learning deep architectures for AI. Foundations and Trends in Machine Learning, Vol. 2, Issue 1, 2009, p. 1-127.

[18] **Tran V. T., Thobiani F. A., Ball A.** An approach to fault diagnosis of reciprocating compressor valves using Teager-Kaiser energy operator and deep belief networks. Expert Systems with Applications, Vol. 41, 2014, p. 4113-4122.

[19] **Tamilselvan P., Wang P.** Failure diagnosis using deep belief learning based health state classification. Reliability Engineering and System Safety, Vol. 115, 2013, p. 124-135.

[20] **Rumelhart D. E., Hinton G. E., Williams R. J.** Learning representations by back-propagating errors. Nature, Vol. 323, 1986, p. 533-536.

[21] **Hinton G. E., Osindero S., Teh Y.** A fast learning algorithm for deep belief nets. Neural Computation, Vol. 18, 2006, p. 1527-1554.

[22] **Hinton G. E., Srivastava N., Krizhevsky A., Sutskever I., Salakhutdinov R.** Improving neural networks by preventing co-adaptation of feature detectors. arXiv 1207.0580, 2012.

[23] **Hinton G. E., Salakhutdinov R. R.** Reducing the dimensionality of data with neural networks. Science, Vol. 313, 2006, p. 504-507.

[24] **Hjelma D., Calhouna V., Salakhutdinov R., Allena E., Adali T., Plisa S.** Restricted Boltzmann machines for neuroimaging: an application in identifying intrinsic networks. Neuroimage, Vol. 96, 2014, p. 245-260.

[25] **Abu-Mahfouz I.** A comparative study of three artificial neural networks for the detection and classification of gear faults. International Journal of General Systems, Vol. 34, Issue 3, 2009, p. 261-277.

[26] **Souza D. L., Granzotto M. H., Almeida G. M., Oliveira-Lopes L. C.** Fault detection and diagnosis using support vector machines – an SVC and SVR comparison. Journal of Safety Engineering, Vol. 3, Issue 1, 2014, p. 18-29.

[27] **Chang C. C., Lin C. J.** LIBSVM: a library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2013.

Zhiqiang Chen received a B.S. degree from WuHan University of Water-Conservancy and Electric Power, WuHan, China and an M.S. degree from ChongQing University, Chongqing, China in 2001 and 2004 respectively. Since 2004, he has been a software engineer in Hanna Strategies Ltd. Shang-hai, China. He received Ph.D. degree from Fukui University, Japan in 2011. He is currently an Associate Professor with the School of Computer Science and Information Engineering, Chongqing Technology and Business University. His main research interests are computational intelligence and signal processing.

Chuan Li received his Ph.D. degree from the Chongqing University, China, in 2007. He has been successively a Postdoctoral Fellow with the University of Ottawa, Canada, a Research Professor with the Korea University, South Korea, and a Senior Research Associate with the City University of Hong Kong, China. He is currently a Professor with the Chongqing Technology and Business University, China, and a Prometeo Researcher with the Universidad Politécnica Salesiana, Ecuador. His research interests include machinery healthy maintenance, and intelligent systems.

René-Vinicio Sánchez received the B.S. in Mechanical Engineering in 2004, from the Universidad Politécnica Salesiana (UPS), Ecuador. He got his Master in management audit quality in 2008 at the UTPL, Ecuador, and the Master degree in industrial technologies research in 2012 at the UNED, Spain. Currently, he is Professor of the Department of Mechanical Engineering in the UPS. His research interests are in machinery health maintenance, pneumatic and hydraulic systems, artificial intelligence and engineering education.