Future Generation Computer Systems 94 (2019) 601–609

Contents lists available at ScienceDirect

Future Generation Computer Systems

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

Multi-scale Dense Gate Recurrent Unit Networks for bearing remaining useful life prediction

Lei Ren [a,b,*], Xuejun Cheng [a,b], Xiaokang Wang [c], Jin Cui [a,b], Lin Zhang [a,b]

[a] School of Automation Science and Electrical Engineering, Beihang University, Beijing, China
[b] Engineering Research Center of Complex Product Advanced Manufacturing System, Ministry of Education, Beijing, China
[c] Department of Computer Science, St. Francis Xavier University, Canada

HIGHLIGHTS

- A new end-to-end deep learning joint optimization method for IOT big data.
- Multi-scale layers, Skip GRU layers and Dense network layers are proposed.
- Demonstrates effectiveness of our MDGRU model based on the true bearing dataset.

ARTICLE INFO

Article history:
Received 31 August 2018
Received in revised form 13 November 2018
Accepted 6 December 2018
Available online 10 December 2018

Keywords:
Internet of things
Smart data
Remaining useful life prediction
Deep learning
Gated Recurrent Unit Network

ABSTRACT

Internet of thing (IoT), with the rapid development, is the systematic combination of physical process, information and communication technologies. Industry internet of thing (IIoT), as the extension of IoT in industry, makes the industrial production more intelligent and efficient. Remaining useful life prediction (RUL), as an essential application area of IIoT, plays an increasingly crucial role. In traditional data-based methods, the feature extraction methods depend on the prior knowledge and are separated from the RUL models. Though ensemble learning can be applied to prevent overfitting, the methods about ensemble learning are still separated from the RUL model. To overcome these drawbacks, a novel deep learning network, namely Multi-scale Dense Gate Recurrent Unit Network (MDGRU) is proposed in this paper, which is composed of the feature layers initialized by pre-trained Restricted Boltzmann Machine (RBM) network, multi-scale layers, skip gate recurrent unit layers, dense layers. By adding multi-scale layers and dense layers, the network can capture the sequence features and ensemble different time-scale attention information. Meanwhile it is an end-to-end network combining the feature extraction methods and RUL models only by pre-training the RBM model so it is more convenient for application. Our experiments with real bearings datasets show that proposed MDGRU network is able to achieve higher accuracy compared to other data-driven methods.

1. Introduction

Internet of thing (IoT) establishes linkages among physical process, information and communication technologies. Internet of thing significantly enhances our daily living by processing the collected smart data. The smart data in the Internet of thing (IoT) provides the new chances and challenges in processing and mining the important information especially in the field of machine health monitoring. However, it is difficult to mining values of smart data in Industry internet of thing due to data size and data complexities. Therefore, there is an emerging need to develop the more efficient data mining techniques in internet of thing (IoT).

As an essential basic element of IoT, bearing plays a crucial role in most industrial machinery. Bearings are one of the most important elements to many machines, which reduce friction between two objects. Any unexpected failure of bearings can lead to negative losses including the reduction of production, extension of downtime and risks of safety. To prevent the failure of bearings, remaining useful life (RUL) prediction is operated to guarantee safety and decrease the loss for the enterprise, which is an indispensable and valuable foundation in IoT [1–4]. Traditional bearing RUL prediction methods often construct models of bearing degradation process based on the assumptions of the linear models or exponential models which are called model-driven methods. These model-driven bearing RUL prediction methods aim to establish mathematic or physical models to describe the bearing degradation

* Corresponding author at: School of Automation Science and Electrical Engineering, Beihang University, Beijing, China.
E-mail addresses: renlei@buaa.edu.cn (L. Ren), chengxuejun@buaa.edu.cn (X. Cheng), xkwang@stfx.ca (X. Wang), jincui@buaa.edu.cn (J. Cui), zhanglin@buaa.edu.cn (L. Zhang).

https://doi.org/10.1016/j.future.2018.12.009
0167-739X/

process and adjust model parameters through measured true data. However, for a complex non-linear system, it is unlikely to build a perfectly accurate mathematic or physical model. In that case, the data-driven models are more effective, which have attracted more and more attention. Data-driven models based on historical industrial big data usually make decisions with the online data from the cloud or edge terminals. Data-driven models usually build particular models based on the machine learning methods which have stronger capabilities for complex non-linear relations. So the data-driven methods typically have stronger modeling capabilities for the data distributions [5]. However, for data-driven methods, it is difficult to find effective features from time domain, frequency domain and time–frequency domain broad range of features. To obtain a more discriminative feature space, new feature extraction methods, such as weighted minimum quantization fusion method [6], unsupervised method [7], ISOMAP method [8], normalization cross correlation indicator method [9] are proposed to easily model the degradation process. Further, applying the above features as inputs, improved remaining useful life (RUL) prediction models constructed, such as Hybrid PSO–SVM-based method [10], Kalman Filter [11], Bayesian Dynamic method [12], etc.

Even so, the above traditional data-driven methods still suffer from the three main shortcomings.

(1) Usually feature extracted methods and RUL models are two parts that are mutual influenced. That means if one part is changed, the other related part must be redesigned. That is complicated and poor universal.

(2) RUL models are trained according to a specific training data and easy overfitting. Although, ensemble learning method can solve the above problems effectively, ensemble learning method usually train many models based on the same data, and set different weight on every models and finally have weighted summation. When model is changes, corresponding weight of model must be redesigned. That is also inconvenient.

(3) For the input data of the RUL models, sequential information is vital for the RUL prediction. But general feature extraction methods only consider the single scale data, which is not enough to capture the information related to the remaining useful life.

The recent advances in Deep Learning (DL) technology have greatly improved the ability to analyze complex data. Additionally, deep learning technology develops to address the need for prediction problems, because it is especially appropriate to highly complex non-linear fitting from artificial neural networks. The deep learning technology provides new challenges for complex prediction problems such as accurate prediction of bearing RUL. In recent years, some new deep learning methods provide some benefits for machine health monitoring area. On the one hand, most researches focus on Deep Neural Network (DNN) based models including Deep Belief Network (DBN) [13], Stacked Auto Encoder (SAE) [14], Auto Encoder Neural Network (AE-DNN) [15], and Restricted Boltzmann Machine Neural Network (RBM-DNN) [16]. These models mostly concentrate on fault diagnosis areas that have less requirements of time series information. On the other hand, CNN and RNN networks have been introduced into the RUL area recently. CNN network has local correlation properties and RNN network has sequence correlation properties [17]. Based on the assumptions that signals about machine health monitoring also have the above two properties, Ruo liu [18] proposed a dislocated time series convolution neural architecture (DTS-CNN) for intelligence fault. Liang Guo [19] adopted the recurrent neural network (RNN) to provide a new health indicator. A. Elsheikh [20] proposed an Long Short-Term Memory (LSTM) network to monitor machine health. The parameters about the CNN and RNN models are usually huge, which are difficult to train and easily overfitting. Multi-scale Dense Gate Recurrent Unit Networks (MDGRU) is a new deep learning network designed to weaken the above concerns. Several state-of-art models are compared with our proposed MDGRU network to verify the effects. The contributions of this paper can be summarized as follows:

(1) Based on the time-domain features, frequency domain features and time–frequency domain features, a new end-to-end joint optimization method, namely Multi-scale Dense Gate Recurrent Unit Network (MDGRU) is proposed, which apply Restricted Boltzmann Machine to handle the high dimensional data and pre-train the model. The feature extraction and RUL models are combined to adjust network parameters jointly.

(2) Multi-scale layers(M) are newly introduced for efficient time-series information representation and Dense (D) network layers are newly introduced to handle the fusion results from the different time scale information outputs. Different scale time information provides different attention features which are contributive to deep learning models.

(3) Skip Gate Recurrent Unit (GRU) layers are designed to enhance the traditional GRU network performance for prediction, which can saves input information during the transforming over the layers and sequential information during the transforming over the time.

(4) Experiments with a real bearing dataset show that our new MDGRU network can achieve improved accuracy compared to other data-driven methods.

This paper is organized as follows. In Section 2, the related Gated Recurrent Unit Network model and Restricted Boltzmann Machine algorithm are presented. And in Section 3, the detailed framework for remaining useful life prediction is proposed. Then, in the following Section 4, the process of remaining useful life prediction is analysis based on the open dataset. Finally, conclusions are drawn in Section 5.

2. Backgroud

2.1. Gated recurrent unit network

Conventional neural network method with a set of inputs for the training examples is to build the relation between input layer x_i and output layer y_i. The recurrent neural network method builds the connection between the above input layer $x_{i-t} \cdots x_i$ and the output layer y_i, which includes feedback activations addressed by the network.

When the input x_t and the corresponding tag o_t, the hidden layer state at time t is determined by the state of the current time t and time $(t-1)$. The relationship between the variables o_t and x_t is shown as follows

$$z_t^h = W_{ih}x_t + W_{hh}h_{t-1} + b_h, \tag{1}$$

$$h_t = f_h(z_t^h), \tag{2}$$

$$z_t^o = W_{ho}h_t + b_o, \tag{3}$$

$$o_t = f_o(z_t^o), \tag{4}$$

where $h = (h_1, h_2 \ldots h_n)$ is the state of the hidden neurons value. It is noted that medium state z_t^h is computed by the connection matrix between the input layer and the hidden layer W_{ih} and the recursive connection matrix inside the hidden layer W_{hh}. And f_h, f_o are the non-linear functions of the output layer. b_h is the bias vector of the hidden layer, b_o is the bias vector of the output layer

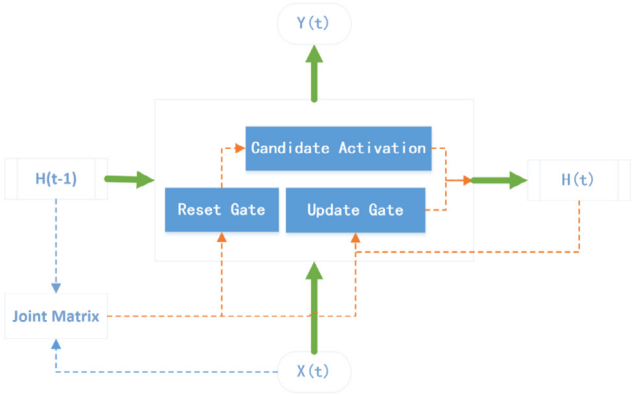

Fig. 1. Traditional GRU model.

When given the input x_t and the corresponding tag o_t, the hidden layer state at time t is determined by the state of the current time$t$ and time $(t-1)$. And there is a dependency about time between the outputs of the different sequences. The study about the error flow [21] shows that the traditional RNN structure can only keep short-term memory because of the vanishing gradient problem. Improved version of the RNN algorithm as Gated Recurrent Unit Network (GRU), can solve the problem about the long time span dependency.

GRU is proposed by Chung [22], as shown in Fig. 1, which captures dependencies of different time scales by recurrent units, which has been successfully applied to sequential or temporal data . The basic structure of GRU is simpler and it has not a memory cell (cell state) compared with RNN. Therefore, the activations of gates in GRU only depend on current input and previous output. These changes make the speed of the GRU model better than that of RNN. And in some cases, the results of the GRU is more accurate than that of RNN. The relationship between input statex_t and update gate z_t is showed as follows:

$$z_t = sigmoid\left(W_{xz}x_t + W_{hz}h_{t-1} + b_z\right), \tag{5}$$

where h_{t-1} represents the hidden states of the time $(t-1)$ and b_z are their biases. The update gate means the degree to update the hidden state information. Also the reset gate r_t means the degree to forget the formal hidden state information.

$$r_t = sigmoid\left(W_{xr}x_t + W_{hr}h_{t-1} + b_r\right). \tag{6}$$

After transforming update gate z_t and reset gate r_t, the hidden state h_t is extended as

$$\tilde{h}_t = tanh\left(W_{xh}x_t + U\left(r_t \odot h_{t-1}\right)\right), \tag{7}$$

$$h_t = \left(1 - z_t\right) \odot h_{t-1} + z_t \odot \tilde{h}_t, \tag{8}$$

where $\odot$ indicates an element-wise multiplication. And if we assume that the final state is y_t, we obtain the output through the following transition function:

$$y_t = sigmoid\left(W_{ho}h_t + b_h\right), \tag{9}$$

where sigmoid means sigmoid function which can be calculated as:

$$\text{sigmoid}\left(x\right) = 1/\left(1 + e^{-x}\right), \tag{10}$$

2.2. Restricted Boltzmann machine

The Restricted Boltzmann Machine (RBM) network is an unsupervised two-layer graph model in deep learning area. The RBM network is composed of visible layers and hidden layers. The nodes on the same layers have no connection, whose purpose is to get lower energy. And minimizing the average negative log-likelihood is applied to train the Restricted Boltzmann Machine (RBM) network .

$$E\left(x, h\right) = -h^T Wx - c^T x - b^T h, \tag{11}$$

$$p\left(x, h\right) = \exp\left(-E\left(x, h\right)\right)/Z, \tag{12}$$

where x is input vector of the network, and h is hidden vector the network,$E\left(x, h\right)$ is energy of the network and we usually use the $p\left(x, h\right)$ to represent the joint probability distribution of the data. The loss function of the RBM is to minimize the following value: [23]

$$\frac{1}{T}\sum_t l\left(f\left(x^{(t)}\right)\right) = \frac{1}{T}\sum_t -\log P(x^{(t)}), \tag{13}$$

RBM has good generalization and mapping capabilities which can approximate any continuous function to solve many complex problems and reduces the dimensionality so as to pre-train the model.

2.3. Enhanced tricks for GRU

Drop out. The equations about the Drop out in RNN showed as follows [24]:

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} sigm \\ sigm \\ sigm \\ tanh \end{pmatrix} T_{2n,4n} \begin{pmatrix} D\left(h_t^{l-1}\right) \\ h_{t-1}^{l} \end{pmatrix}, \tag{14}$$

where the dropout operator D is used to set a random subset in hidden state h_t^{l-1} to zero. The dropout operator corrupts the information carried by the units, also save the partial information. By the above methods, we can achieve the information flow transformation from the former time step to the current time step.

Activation Function. The formulas of the activation functions showed as follows:

$$Relu\left(x\right) = \max\left(0, x\right), \tag{15}$$

$$\text{Sigmoid}\left(x\right) = 1/\left(1 + e^{-x}\right). \tag{16}$$

Relu activation function and Sigmoid activation function can help solve the problem of the gradient dispersion, also it can prevent over-fitting.

Adam optimization algorithm. Adam algorithm was firstly proposed by Diederik Kingma [25]. The main benefits of using Adam algorithm on convex optimization problem are invariant to diagonal rescale of the gradients and little memory requirements. The Adam algorithm combines the advantages of Adaptive Gradient Algorithm (AdaGrad) and Root Mean Square Propagation (RMSProp) [26,27].

In our proposed framework, Multi-scale Dense Gate Recurrent Unit Networks are proposed with multi-scale layers, skip layers and dense layers. Restricted Boltzmann Machine is applied to increase the efficiency of the model training. Multi-scale Dense Gate Recurrent Unit Networks enhance the feature learning skills, which can improve the expressiveness of the GRU model.

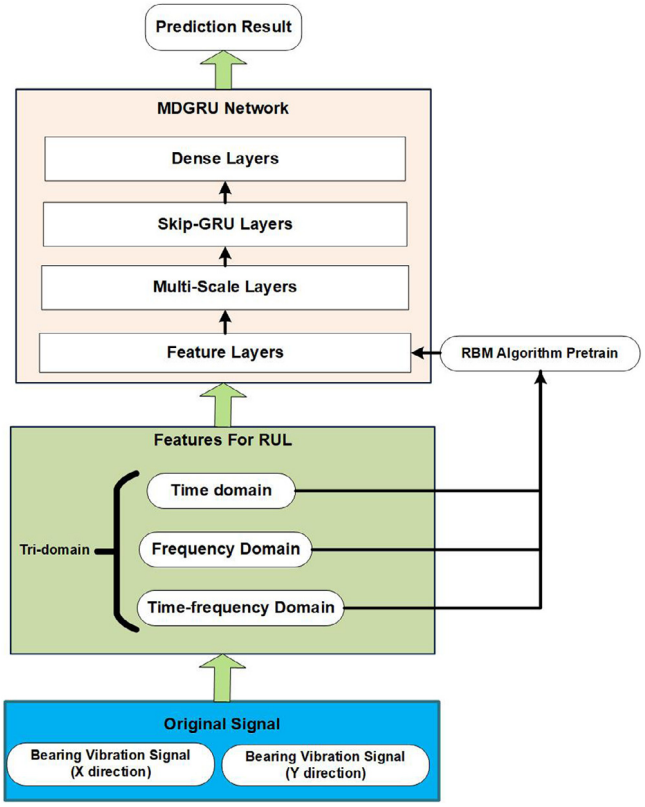

Fig. 2. Framework of remaining useful life prediction.

3. The proposed framework

3.1. RUL process

To demonstrate the complete framework for bearing remaining useful life prediction, as shown in Fig. 2, the detailed process of remaining useful life prediction (RUL) can be summarized as follows:

Step1. Preprocessing. The measure data from the rolling bearing is collected and the healthy condition indicator for the remaining useful life is defined. The remaining useful life of a rolling bearing means the available useful time from current moment to the failed moment [28].

Step2. Tri-domain feature extraction. Three domain features including time-domain features, frequency-domain features and time–frequency-domain features are extracted for signals. The features are feed into the Restricted Boltzmann Machine network (RBM) to pre train the MDGRU network. To enable the feature embedding, these features are normalized to the range $[-1, 1]$.

*Step*3. Deep learning model establishment. The feature layers initialized according to RBM network. Other layers initialized randomly. The parameters including the network structure, different time scale values, the number of the neuron, the dropout parameters are adjusted. MDGRU network consists of two-layer feature layers, one-layer multi-scale layer, several skip-GRU layers and three-layer dense layers.

Step 5. Model prediction. The final MDGRU network model trained well is applied to predict the remaining useful life and obtain the prediction results.

3.2. Multi-scale dense gate recurrent unit network

RNN has been used successfully in many areas such as Image Captioning, Speech Recognition, Machine Translation, Computer composed Music. Multi-scale Dense Gate Recurrent Unit Networks is an improved network of the Gate Recurrent Unit Network. The signals collected by the sensors usually are time series signals with strong periodicity and deep correlations among different time points [29]. The scale of time series are difficult to be decided. If the short scale is considered, the result may leave out the important information. On the meanwhile, if too long scale is computed, the network is difficult to train and may cause the waste of time and resources. To solve the above problems, we proposed the MDGRU network to save the enough information of the features and train the network conveniently based on the IoT big data. As shown in Fig. 3, MDGRU network constructs with four type layers: feature layers, multi-scale layers, skip-GRU layer and dense layers.

In MDGRU network, the feature layers are two layers neural network. Its initial parameters are from the Restricted Boltzmann Machine (RBM) trained model. Other parameters about the MDGRU except the feature layers initialize randomly. The multi-scale layers accept the feature layers output and choose different time scale information to several models composed of skip GRU layers, which learn the time sequence information. The outputs of several models composed of skip GRU layers outputs are concatenated one output. Dense layers accept the one output and finally output the remaining useful life.

3.2.1. Feature layers

One the one hand, a few features may lose the important information, on the other hand, excessive features as input will lead to a large network model and may cause the waste of time. So enough tri-domain features are extracted and Restricted Boltzmann Machine network are used to reduce the dimension of tri-domain features. Furthermore, the Restricted Boltzmann Machine (RBM) algorithms only provide the initial parameters for feature layers, so the network is more robust even if the RBM model trained not well.

3.2.2. Multi-scale time layers

Word embedding, which map the words or phrased to vectors of specific numbers, has been used widely in natural language processing (NLP). And there are many different embedding methods to implement the embedding function [26]. In the RUL area, to meet the demands about GRU network input, the multi-scale time layers are applied to encode the feature to vector, which can prevent the loss of the information of the feature. The basic formula shows as follows:

$$D(a, n, :) = x(t_{ai} : t_{aj}), \tag{17}$$

where

$$t_{ai} = a * scale, \tag{18}$$

$$t_{aj} = t_{ai} + scale, \tag{19}$$

$$a = 0, 1, 2 \ldots m,$$

where $D \in R^{a*n*d}$ is an matrix as the output of feature, n is the step, and d is the length of the signal feature. There is not an accurate method to decide the values of scale. The scale values usually are decided according to the data size of the model and the dimensions of the features.

3.2.3. Skip GRU layers

The enhanced strategies make the GRU network perform better than traditional simple GRU network, which includes the drop out strategy, Relu activation function, Adam Optimization algorithms. Considering information in the middle of the sequence might be easily lost in the GRU, the skip connection is added to the model, which is main difference between our network and traditional GRU network.

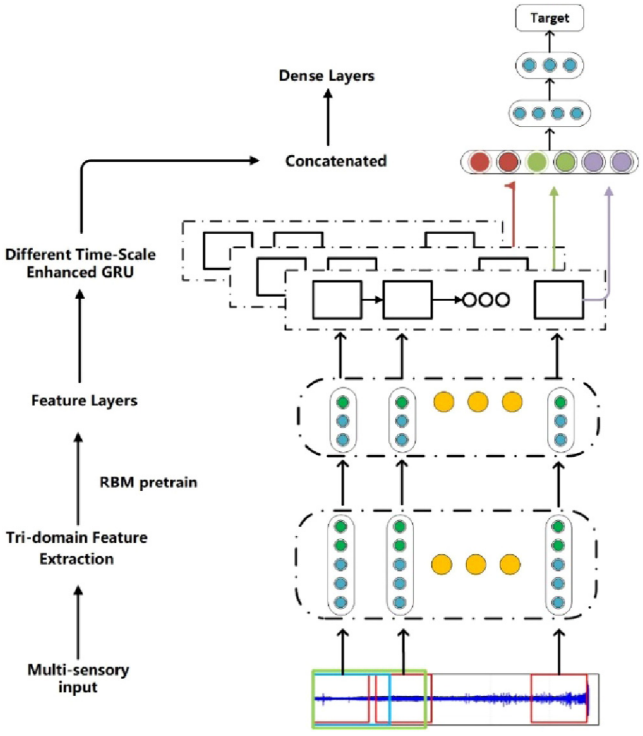

Fig. 3. MDGRU network.

Skip Connection: To enhance the capacity of network to learn the complex non-linear relationships, the network can be designed to the multi GRUs, but the information among the network transmitting might be lost in traditional GRU network. To overcome the shortcomings and guarantee the enough information remaining. The skip connection is imported to the GRU model as shown in Fig. 4. The former input of GRU network is x_i, and the output of first GRU network is y_i. The relationship about the input and output is

$$h_{ij} = H_j(x_{ij}), \tag{20}$$

where the function H_j is defined in formula (8) , and j is the number order of GRU network in multi-GRUs model, and the skip connection is processed by adding the input of the former GRU by the output of the former GRU.

$$x_{i(j+1)} = h_{ij} + x_{ij}. \tag{21}$$

The next GRU network formula showed as follows:

$$h_{i(j+1)} = H_{j+1}(x_{i(j+1)}). \tag{22}$$

We design the skip-GRU layers of the different time scale with the same network structure. Because the parameters of the network are random initialized. The parameters have the same network structure but the parameters are different, which makes different skip-GRU layers can have attention on the different time scale information.

3.2.4. Dense layers

Ensemble learning use to handle complex relations among the noisy data. Existing works on ensemble learning on Prognostics and Health Management (PHM) area include the pareto neuro-evolution [30] and diverse and accurate ensemble learning [31]. These bearing signals largely vary with a number of factors including different time scale information. Many works have demonstrated that any of the single learning algorithms seldom perform consistently well. A single GRU system may not capture varying information completely, thus giving rise to the need for Dense layers.

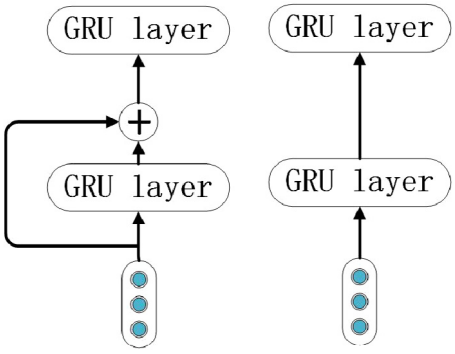

Fig. 4. GRU layers with skip connection and no skip connection.

However, many ensemble approaches ensemble the results of the many well-trained models. The approaches will separate ensemble learning methods from the network training process, which may lead to lower generalization performance and waste of the training time. To solve the problem, dense layers are added to the network, which make the model trained jointly. The Dense layer is 3 layers neural network, which ensembles three time scale output.

The input of dense layers is as follows:

$$x_i = h_{1i} \oplus h_{2i} \oplus h_{3i}, \tag{23}$$

where dense layer layers input x_i concatenate the three time scale enhanced GRU output h_{1i}, h_{2i}, h_{2i}.

4. Application on FEMTO-ST System

To investigate the effectiveness of the MDGRU method, a numerical experiment is implemented based on the FEMTO-ST data. To validate the effectiveness of our proposed method, experiments of different environments and working conditions are conducted.

4.1. Date description

The data provided by the FEMTO-ST institute in France are from the IEEE PHM2012[1] Predictor Challenge and are used to verify the effectiveness of our proposed methods. The bearing data are obtained from 17 sets of bearings in the case of accelerated life experiments. In the process of the experiments, two accelerometers are used in the vibration signal acquisition both for horizontal and vertical directions in the bearing. The load is from horizontal direction. Data sampling frequency is 25.6 kHz and the data is collected every ten seconds. And the time of collecting data is 0.1 s. When the accelerometer exceeds 20 g, the bearing is considered to be defective.

There are 17 sets of data obtained under three operating conditions. All bearings are operated from test conditions to failure. Because the remaining useful life can only be measured when the bearing breaks down. This experiment needs to use bearing data of failure as training data. The test data need other bearing data. Signal at some point from the above bearing is from 2560 sampling points. Every sampling points collect data in two directions. To reduce random effect, each algorithm is executed for many times.

[1] http://data-acoustics.com/measurements/bearing-faults/bearing-6/

4.2. Tri-domain feature

Our tri-domain features include time-domain features, frequency-domain features and time–frequency-domain features.

In the time domain, the statistical features are calculated including maximum value, minimum value, absolute mean value, peak values, RMS, mean value, standard deviation, skewness, kurtosis, variance, waveform factor, crest factor, coefficient of variation, skewness coefficient, coefficient of kurtosis, clearance factor, pulse factor, energy operator, root mean, clearance factor. In the time domain, the above 20 metrics are calculated and 40 features are obtained considering two directions.

In the frequency domain, the FFT method is applied to extract the frequency-domain feature. In the frequency domain, six subbans frequency spectra based on the above FFT method are lead in and finally 12 features are acquired because of two directions.

In time–frequency domain, eight energy ratios generating by three-level wavelet packet decomposition are added into the input features and finally 16 features are obtained because of two directions.

After acquiring the 68 features according to the above formula, the Multi-scale Dense Gate Recurrent Unit (MDGRU) network then is applied to predict the remaining useful life (RUL) of the bearing.

4.3. MDGRU architecture

Based on the 68 features extracted from the feature extraction architecture, the feature layers are used to convert the input 68 features to 34 features. The initialized parameters of feature layers are provided by The Restricted Boltzmann Machine (RBM) algorithms. The detailed information about RBM is that the epoch is 40, learning rate 0.01 and the batch size is 64. And the multi-scale layers are then introduced into the network. The main parameters in the multi-scale layers are time scale and its value are comparatively difficult to determine, three scales of the data are chose to make sure that we take more information into consideration.

Considering the limited number of the feature and training data, the number of hidden layer is set in [3,5] in our experiments. The three time scales are set to 3, 5, 10 respectively, also structure of the different scale network is the same. For the time scale t, the input dimension is $(t, 34)$, 34 means the number of the tri-domain feature. Then the skip connection is transmitted between the connected two layers. Three time scales output vector sizes are both $(1, 10)$. The activation of the hidden layers is sigmoid function and the activation of the final output layer is Relu function. And the drop-out layers are set as 0.4. Also the variables of the network are optimized by Adam. The variables of the training process are: 500 for batch size, 50 for training epochs.

In the Dense layers, the size of input vector is $(1, 30)$, which concatenate the three outputs of different scales network. And three MLP network are constructed as the dense layers. The final outputs are passed into the sigmoid function so the value is restricted into $(0, 1)$. The remaining useful life whose value is in $(0, 1)$ is defined as the available useful time from current moment to the moment if failed. The upper limit of RUL are set in 2810 and RUL for all bearings will be divided by 2810.

To evaluate the accuracy of the RUL prediction, RMSE metrics are introduced in the experiments. RMSE is commonly used to evaluate prediction accuracy, which has equal weight both early and late prediction on the whole life of the bearing. The formula of this RMSE is as follows:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N} d_i^2} \tag{24}$$

where d_i is the absolute error of the ith error, and N is the result of the RUL predictions.

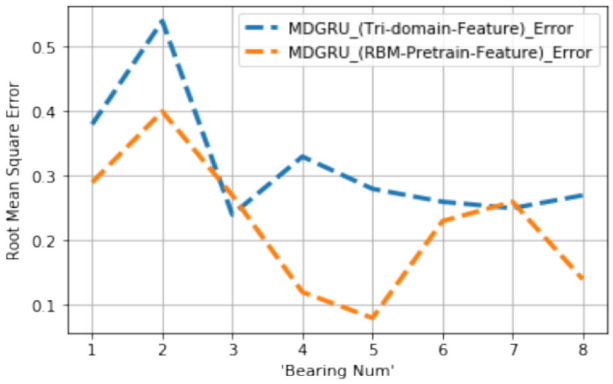

Fig. 5. The comparison under different feature extraction methods.

To verify the effectiveness of the MDGRU network, different approaches are compared with our models: SVM (Support Vector Machine/Regression) uses time-series structures to model the bearing degradation. SVM algorithm is implemented in the RUL prediction . Random Forest makes predictions based on the branches of decision trees [32]. Multi-Layer Perceptron (MLP) is the classical neural network, which learning the functional relationship between input and output through hidden layers. Deep MLP has more learning ability about the feature [33]. Bayesian regression is proposed to carry out the linear regression based on the Bayesian inference. Also, linear Regression and Decision Tree are constructed as the comparative example. Each method in comparison is run for ten times and we got the average value to report.

4.4. Experimental results

In this paper, a new deep learning model, namely MDGRU network, is proposed using different time-scale feature information to improve the model accuracy. In order to show the advantages of MDGRU deep learning network, the MDGRU network is compared with the traditional machine learning algorithms such as SVM, Random Forest, Bayesian Regression algorithm and so on. The results illustrate the advantages of the proposed MDGRU network.

We implement SVM models using *LibSVM*. Also, Bayesian Regression, Decision Tress, Random Forest, Linear Regression are constructed with *Sklearn*. We implement MDGRU, GRU and DNN using *Tensorflow* [34].

It is obviously indicated that the MDGRU using the features with RBM is more effective than the features without RBM from Fig. 5. So the features using RBM are more suitable for our deep learning model than traditional tri-domain features.

MDGRU network is propose based on the GRU network. It is necessary to validate the advantages of MDGRU by comparing the tradition GRU network with our MDGRU network using similar data. It can be seen in Fig. 6, six bearing test data among all eight bearing test data show that MDGRU models get better prediction accuracy than shallow GRU model. The main reason is that traditional GRU network only adaptively captures the representative information from the specific time scale feature, which means the information from the features is not enough. MDGRU network ensembles three different time scale information, which is more reliable for learning robust characteristics from the measured vibration signals. So our proposed MDGRU network is more effective than traditional GRU network.

To verify the accuracy of MDGRU network, the experimental results processing by different algorithms are shown in TABLE II and in Fig. 7. The RMSE in Table 1 is calculated based on the average value about the bearings NO. 1–3, 1–4, 1–5, 1–6, 2–3, 2–4, 2–5 and

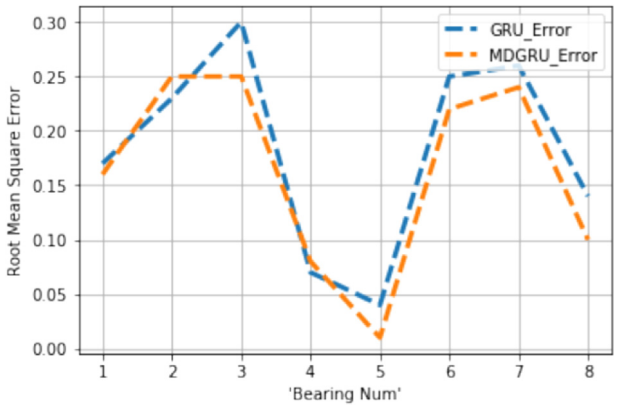

Fig. 6. The comparison between MDGRU and GRU model.

2–6. These bearings are running under different operating environmental conditions. Therefore, the RUL prediction is typically more difficult than traditional single bearing RUL prediction, and it is more likely to produce greater errors. But the results from Table 1 show that, even in this case, MDGRU network still achieves better results with lowest RMSE.

The related parameters of the other methods are described as follows. (1) DNN: the structure of DNN is 34-20-10-1. The learning rate, iteration are 0.05, 500, respectively. (2) Random Forest: the criterion is 'gini' value and max_depth is 6. (3) Bayesian Regression: the number of iterations is 300 and shape parameter for the Gamma distribution prior over the alpha is 1.e−6. (4) SVM: RBF kernel is chosen. And the penalty factor and the radius of the kernel function are 25 and 0.08 respectively. (5) Linear Regression: the normalize value is set to false. (6) Decision Tree: the max_depth is 3 and the minimum number of a left node are chosen as 1.

To sum up, our proposed method is compared with several state-of-the-art methods in bearing remaining life prediction. The first observation is that our proposed MDGRU, which encode different time scale information on the features, performs better than traditional data-driven models including SVM, Random Forest and Bayesian Regression and the deep neural networks. Therefore, the importance of modeling the time scale dependency in remaining useful life prediction has been verified. Furthermore, our proposed MDGRU network outperforms traditional GRU model because of the introduction of multi-scale reconstruction layer and dense layers. As shown in Table 1, the robust performances achieved by our proposed method in the above remaining life prediction tasks verify the generalization capability of our method.

Table 1
The results of mdgru and machine learning algorithms.

Algorithm	RMSE
MDGRU	**0.152**
DNN	0.184
Random Forest	0.209
Bayesian Regression	0.198
SVM	0.211
Linear Regression	0.2
Decision Tree	0.188

5. Conclusions

In this paper, a new deep learning method MDGRU is proposed to predict bearing remaining useful life. To evaluate our method, real experimental rolling bearing PHM datasets are provided to verify the effectiveness and efficiency of the new method. The results show the superiority of MDGRU. Through the comparison with traditional machine learning methods, it is reasonable to conclude that the MDGRU network has three critical features:

(1) MDGRU is an end-to-end network, which is less dependent on prior knowledge. In addition, it is feasible for large-scale industrial data, and the performance is improving with the increase of the size of datasets.

(2) *Multi-scale* layers and *skip GRU* layers are proposed in MDGRU network. The multi-scale layers and dense layers make sense by ensemble learning, So MDGRU layers save enough time information and prevent overfitting.

(3) The new MDGRU network can achieve higher accuracy compared to other data-driven methods. And the significant performance of MDGRU expands the application range of RNN deep learning method.

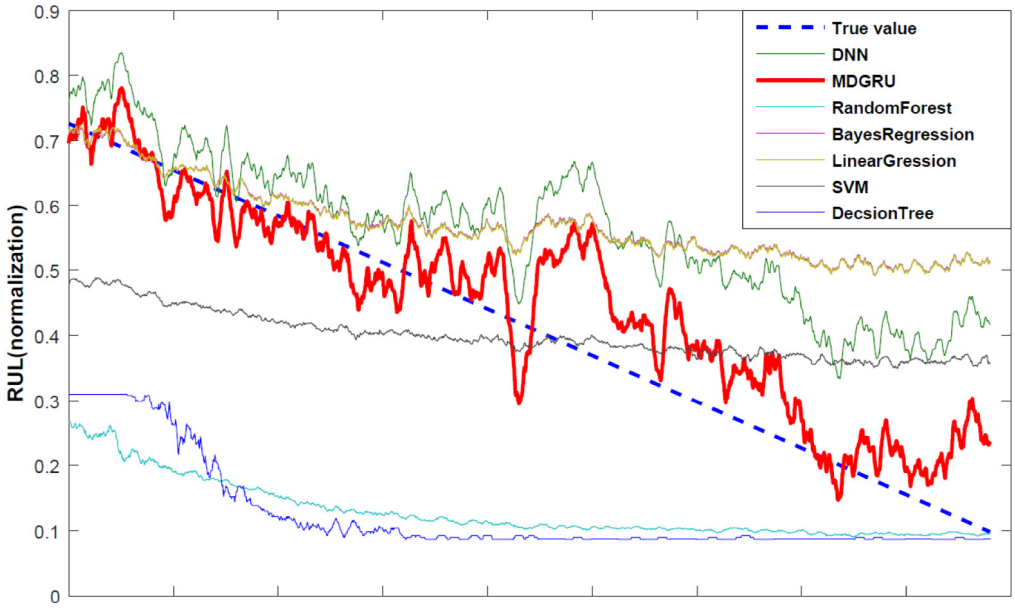

Fig. 7. The comparison between different algorithms.

We expect that the new deep learning method can be easily adapted to other industrial big data applications, which is in progress within our group. For future steps, we also plan to further explore other algorithms such as the *attention model* method.

Acknowledgments

The research is supported by The National Key Research and Development Program of China No. 2018YFB1004001, the NSFC (National Science Foundation of China) project No. 61572057 and 61836001 and the Academic Excellence Foundation of BUAA, China for Ph.D. Students.

References

[1] J. Lee, B. Bagheri, H.-A. Kao, A cyber-physical systems architecture for industry 4.0-based manufacturing systems, Manuf. Lett. 3 (2015) 18–23.
[2] X. Wang, L.T. Yang, H. Liu, M.J. Deen, A big data-as-a-service framework: State-of-the-art and perspectives, IEEE Trans. Big Data 4 (2018) 325–340.
[3] X. Wang, L.T. Yang, X. Xie, J. Jin, M.J. Deen, A cloud-edge computing framework for cyber-physical-social services, IEEE Commun. Mag. 55 (2017) 80–85.
[4] J. Lee, H.D. Ardakani, S. Yang, B. Bagheri, Industrial big data analytics and cyber-physical systems for future maintenance & service innovation, Procedia CIRP 38 (2015) 3–7.
[5] R. Razavi-Far, M. Farajzadeh-Zanjani, M. Saif, An integrated class-imbalanced learning scheme for diagnosing bearing defects in induction motors, IEEE Trans. Ind. Inf. 13 (2017) 2758–2769.
[6] Y. Qian, R. Yan, R.X. Gao, A multi-time scale approach to remaining useful life prediction in rolling bearing, Mech. Syst. Signal Process. 83 (2017) 549–567.
[7] A. Mosallam, K. Medjaher, N. Zerhouni, Data-driven prognostic method based on Bayesian approaches for direct remaining useful life prediction, J. Intell. Manuf. 27 (2016) 1037–1048.
[8] T. Benkedjouh, K. Medjaher, N. Zerhouni, S. Rechak, Remaining useful life estimation based on nonlinear feature reduction and support vector regression, Eng. Appl. Artif. Intell. 26 (2013) 1751–1760.
[9] Q. Zhang, P.W.-T. Tse, X. Wan, G. Xu, Remaining useful life estimation for mechanical systems based on similarity of phase space trajectory, Expert Syst. Appl. 42 (2015) 2353–2360.
[10] P.G. Nieto, E. Garcia-Gonzalo, F.S. Lasheras, F.J. de Cos Juez, Hybrid PSO–SVM-based method for forecasting of the remaining useful life for aircraft engines and evaluation of its reliability, Reliab. Eng. Syst. Saf. 138 (2015) 219–231.
[11] M. Bressel, M. Hilairet, D. Hissel, B.O. Bouamama, Remaining useful life prediction and uncertainty quantification of proton exchange membrane fuel cell under variable load, IEEE Trans. Ind. Electron. 63 (2016) 2569–2577.
[12] F. Sun, N. Wang, X. Li, W. Zhang, Remaining useful life prediction for a machine with multiple dependent features based on Bayesian dynamic linear model and copulas, IEEE Access 5 (2017) 16277–16287.
[13] C. Zhang, P. Lim, A. Qin, K.C. Tan, Multiobjective deep belief networks ensemble for remaining useful life estimation in prognostics, IEEE Trans. Neural Netw. Learn. Syst. 28 (2017) 2306–2318.
[14] 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.
[15] H. Shao, H. Jiang, H. Zhao, F. Wang, A novel deep autoencoder feature learning method for rotating machinery fault diagnosis, Mech. Syst. Signal Process. 95 (2017) 187–204.
[16] H. Shao, H. Jiang, X. Zhang, M. Niu, Rolling bearing fault diagnosis using an optimization deep belief network, Meas. Sci. Technol. 26 (2015) 115002.
[17] L. Ren, Y. Sun, H. Wang, L. Zhang, Prediction of Bearing Remaining Useful Life With Deep Convolution Neural Network, IEEE Access 6 (2018) 13041-9.
[18] 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. Ind. Inf. 13 (2017) 1310–1320.
[19] L. Guo, N. Li, F. Jia, Y. Lei, J. Lin, A recurrent neural network based health indicator for remaining useful life prediction of bearings, Neurocomputing 240 (2017) 98–109.
[20] A. Elsheikh, S. Yacout, M.-S. Ouali, Bidirectional handshaking LSTM for remaining useful life prediction, Neurocomputing (2018).
[21] R. Rana, Gated recurrent unit (gru) for emotion classification from noisy speech, arXiv preprint arXiv:1612.07778, 2016.
[22] J. Chung, C. Gulcehre, K. Cho, Y. Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling, arXiv preprint arXiv:1412.3555, 2014.
[23] M. Tanaka, M. Okutomi, A novel inference of a restricted Boltzmann machine, in: Pattern Recognition, ICPR, 2014 22nd International Conference on, IEEE, 2014, pp. 1526–1531.
[24] Y. Gal, Z. Ghahramani, A theoretically grounded application of dropout in recurrent neural networks, in: Advances in Neural Information Processing Systems, 2016, pp. 1019–1027.
[25] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980, 2014.
[26] O. Levy, Y. Goldberg, Neural word embedding as implicit matrix factorization, in: Advances in Neural Information Processing Systems, 2014, pp. 2177–2185.
[27] W.-S. Chin, Y. Zhuang, Y.-C. Juan, C.-J. Lin, A learning-rate schedule for stochastic gradient methods to matrix factorization, in: Pacific-Asia Conference on Knowledge Discovery and Data Mining, Springer, 2015, pp. 442–455.
[28] X. Wang, L.T. Yang, L. Kuang, X. Liu, Q. Zhang, M. Jamal Deen, A tensor-based big data-driven routing recommendation approach for heterogeneous networks, IEEE Netw. Mag., https://doi.10.1109/MNET.2018.1800192.
[29] D. Pan, J.-B. Liu, J. Cao, Remaining useful life estimation using an inverse Gaussian degradation model, Neurocomputing 185 (2016) 64–72.
[30] H.A. Abbass, Pareto neuro-evolution: Constructing ensemble of neural networks using multi-objective optimization, in: Evolutionary Computation, 2003 CEC'03 the 2003 Congress on, IEEE, 2003, pp. 2074–2080.
[31] A. Chandra, X. Yao, Ensemble learning using multi-objective evolutionary algorithms, J. Math. Model. Algorithms 5 (2006) 417–445.
[32] 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 Process. 76 (2016) 283–293.
[33] L. Ren, J. Cui, Y. Sun, X. Cheng, Multi-bearing remaining useful life collaborative prediction: A deep learning approach, J. Manuf. Syst. 43 (2017) 248–256.
[34] L. Ren, Y. Sun, J. Cui, L. Zhang, Bearing remaining useful life prediction based on deep autoencoder and deep neural networks, J. Manufacturing Syst. (2018).

Lei Ren is an associate professor and the deputy head of Cloud Manufacturing Research Center at School of Automation Science and Electrical Engineering in Beihang University, and a senior research scientist at Engineering Researching Center of Complex Product Advanced Manufacturing Systems, Ministry of Education, China. He received a Ph.D. degree in computer science from the Institute of Software, Chinese Academy of Sciences in 2009. His research interests include big data analytics and applications. He has published 50+ papers and got 2000+ citations according to Google Scholar. He edited a book entitled "Challenges and Opportunity with Big Data" published by Springer LNCS. He served as an associate editor of SIMULATION: Transactions of the Society for Modeling and Simulation International, and reviewers for journals such as IEEE TII, TSMC, ACM TCC. He also served as a TPC co-chair for the 19th Monterey Workshop on Big Data, 5th International Conference on Enterprise Systems, etc. He is a member of IEEE, ACM, ASME and SCS.

Xuejun Cheng received the B.Eng. degree in automation engineering from Beihang University, China, in 2016, where he is currently pursuing the M.S. degree with the School of Automation Science and Electrical Engineering. His research interests include deep learning, machine learning for industrial big data analytics about manufacturing.

Xiaokang Wang received the B.S. degree in Electronic and Information Engineering from Henan Normal University, Xinxiang,China, in 2009, the M.S. degree in Computer Science from Changzhou University, Changzhou, China, in 2012, and the Ph.D degree in Computer System Architecture in Huazhong University of Science and Technology,Wuhan, China, in 2017. Currently,he is a Postdoctoral Research Fellow of Department of Computer Science, St. Francis Xavier University, Canada. His research interests are Cyber–Physical–Social Systems, Big Data, Tensor Computing, and Parallel and Distributed Computing.

Jin Cui is currently pursuing the Ph.D. degree with the School of Automation Science and Electrical Engineering, Beihang University, China. He received the B.Eng. degree in automation from Taiyuan University of Technology, China, in 2013. His current research interests include prognostic and health management, trust management in cloud manufacturing.

Lin Zhang is a full professor of Beihang University. He received the B.S. degree in 1986 from the Department of Computer and System Science at Nankai University, China, the M.S. degree and the Ph.D. degree in 1989 and 1992 from the Department of Automation at Tsinghua University, China. From 2002 to 2005 he worked at the US Naval Postgraduate School as a senior research associate of the US National Research Council. His research interests include complex systems modeling and simulation, data science, and model engineering for simulation. He serves as the Past President of the Society for Modeling & Simulation International (SCS), a vice president of the Chinese Simulation Federation (CSF), Fellow of ASIASIM, a board member of CAAI, a senior member of IEEE, a member of IEEE TC on Industrial Informatics, the chief scientist of key projects of China High-Tech R&D Program (863), and associate Editor-in-Chief and associate editors of 6 peer-reviewed international journals. He authored and co-authored 200 papers, 10 books and chapters. He received the National Award for Excellent Science and Technology Books in 1999, the 863 Outstanding Individual Award in 2001, the National Excellent Scientific and Technological Workers Awards in 2014.