IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 1, 2019, accepted September 20, 2019, date of publication September 24, 2019,
date of current version October 9, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2943381

AESGRU: An Attention-Based Temporal Correlation Approach for End-to-End Machine Health Perception

WEITING ZHANG[1], DONG YANG[1], (Member, IEEE), HONGCHAO WANG[1], (Member, IEEE), JUN ZHANG[2], AND MIKAEL GIDLUND[3], (Senior Member, IEEE)

[1] School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing 100044, China
[2] Beijing Sheenline Technology Company Ltd., Beijing 100044, China
[3] Information and Communication Systems, Mid Sweden University, 851 70 Sundsvall, Sweden

Corresponding author: Dong Yang (dyang@bjtu.edu.cn)

This work was supported in part by the Fundamental Research Funds for the Central Universities under Grant 2018YJS013, in part by the National Key Research and Development Program of China under Grant 2018YFB1702001, and in part by the National Natural Science Foundation of China under Grant 61771040.

ABSTRACT Accurate and real-time perception of the operating status of rolling bearings, which constitute a key component of rotating machinery, is of vital significance. However, most existing solutions not only require substantial expertise to conduct feature engineering, but also seldom consider the temporal correlation of sensor sequences, ultimately leading to complex modeling processes. Therefore, we present a novel model, named Attention-based Equitable Segmentation Gated Recurrent Unit Networks (AESGRU), to improve diagnostic accuracy and model-building efficiency. Specifically, our proposed AESGRU consists of two modules, an equitable segmentation approach and an improved deep model. We first transform the original dataset into time-series segments with temporal correlation, so that the model enables end-to-end learning from the strongly correlated data. Then, we deploy a single-layer bidirectional GRU network, which is enhanced by attention mechanism, to capture the long-term dependency of sensor segments and focus limited attention resources on those informative sampling points. Finally, our experimental results show that the proposed approach outperforms previous approaches in terms of the accuracy.

INDEX TERMS Health perception, temporal correlation, gated recurrent unit networks, long-term dependency, attention mechanism.

I. INTRODUCTION

Recently, the prognostics and health management (PHM) system has become a reliable solution for managing the health status of industrial machinery (e.g., predictive maintenance, PdM) [1]. For rotating machinery, accurate and real-time perception of operating status of rolling bearing is of vital significance [2], which can effectively avoid catastrophic failures and minimize maintenance costs of industrial manufacturing. Therefore, it is necessary to accurately identify faults and perform maintenance in the most effective manner [3]. With the rapid development of smart sensors [4], signal processing and artificial intelligence (AI), data-driven methods have gradually become the mainstream solution for the PdM [5], and is widely applied to perform fault diagnosis of industrial equipment.

During the past decades, traditional machine learning (ML) models has received great success in various applications, including the health perception [6]. Some algorithms, such as support vector machine (SVM) [7], [8], random forests (RF) [9], and regression model [10], have achieved remarkably results. But notably, as shown in **Figure 1**, ML algorithms generally require feature engineering to extract important features, which may result in additional human labor and substantial expertise to complete efficiently.

As the increasing number of deployed sensors, the volume of industrial data is growing dramatically [11]. But with the increase in computational power, and continuous innovation in algorithms, deep learning (DL) [12] has demonstrated tremendous potential [13]. It can learn more complex patterns using deep hidden layers between the input and output, at the

The associate editor coordinating the review of this manuscript and approving it for publication was Dong Wang.

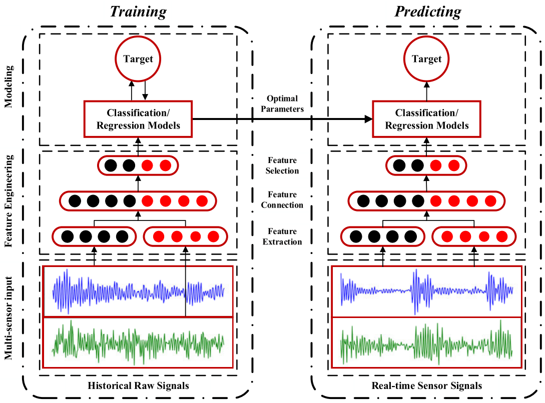


FIGURE 1. Data-driven method for health perception of industrial equipment.

same time reduces the algorithm's dependence on feature engineering, as stated in [14]–[16].

As is known, there are two main-stream structures of neural networks, namely convolutional and recurrent structure. In particular, the recurrent based models are especially good at representing the temporal characteristic of sequential data. In practice, the sensor signals record the equipment status in the form of time-series, and its internal dependence is of great importance to the diagnostic effect.

Many recent works have demonstrated that RNN models, can effectively deal with time-series signals. Remarkably, the LSTM is the most widely used variant of RNN. For instance, Zhao *et al.* [17] focused on predicting actual tool wear conditions using long-short term memory (LSTM). Similarly, an LSTM-based method [18] was utilized to carry out a RUL prediction of an aero-engine. In [19], a convolutional bidirectional LSTM network was proposed to predict the actual wear of a high-speed CNC machine. Notably, this paper integrated two mainstream structures of neural network (i.e., convolutional and recurrent). In addition, an LSTM-based encoder-decoder framework was deployed to obtain an unsupervised health indicator from multi-sensor time-series data [20]. Although the LSTM has made a lot of exploration in the area of PdM, it also needs more efforts in the aspect of computational efficiency. Thus, the GRU network came into being and obtained the preliminary application. For instance, a hybrid approach called LFGRU, which combines automatic feature learning with handcrafted features, was proposed to perform machine health monitoring [21]. This work has greatly improved the prediction metrics of GRU network. But totally, although it has achieved remarkable results in the fault diagnostic, there are still considerable works with regard to optimizing model complexity and prediction accuracy of algorithm.

Except the temporal characteristic, the time-series such as vibration signal also exist the correlation between sampling points. In this paper, to enable RNN models better capturing information from the entire signal sequence, we design a novel model, called AESGRU, to represent the raw signal via an end-to-end approach, and the attention mechanism has been introduced to balancing the temporal correlation and computational efficiency. Our contributions are summarized as follows:

1) Our proposed approach does not require any feature engineering and completely implements an end-to-end diagnostic system. Considering the inherent properties of the sensor sequence, an equitable segmentation method of multi-sensory is proposed. In specific, we divide the original sensor dataset into equal-length segments, so that the temporal correlation of the complete sequence can be integrally preserved. And these newly generated samples will be fed into the model sequentially based on their respective identifiers.
2) We deploy a single-layer bidirectional GRU, improved with attention mechanism, to represent and learn the strongly correlated equally-segment. Notably, the attention mechanism is introduced to directly extract the potential relationship of discrete sampling points from the original sensor signal, which significantly contributes to the capture of long-term dependency for target prediction (i.e., the equipment operation status) and focus limited attention resources on those informative sampling points. Besides, it is also helpful to improve model interpretability, and provides a reliable means to improve modeling efficiency.
3) Moreover, our model is also suitable for multi-sensor scenarios, which can capture the temporal correlation form the two-dimensional vibration signal simultaneously. Ultimately, we conduct sufficient experiments on the Case Western Reserve University (CWRU) dataset. The validity of our model has been verified with regard to its accuracy, time efficiency, confusion matrix and attention weight distribution.

The remainder of this paper is organized as follows. In Section II, several related algorithms are introduced. In Section III, our proposed AESGRU is described. In Section IV, the experiments on the CWRU dataset are conducted. Finally, concluding remarks are provided in Section V.

II. RECURRENT NEURAL NETWORKS AND ATTENTION MECHANISM

A. RECURRENT NEURAL NETWORKS AND ITS VARIANTS

The RNN is especially utilized to process one-dimensional sequential data. The time step index of this does not necessarily literally correspond to time elapsed in the real world. It may simply represent the position in the sequence. As shown in **Figure 2**, the key aspect of RNN is that it can be used to connect previous information to the current state [22]. The structure of its hidden units is shown in **Figure 3(a)** and the hidden state is computed by Eq. (1):

$$h_t = f\left(h_{t-1}, x_t\right) \tag{1}$$

where f is a nonlinear activation function, usually the *sigmoid*, *tanh* or *ReLU* unit.

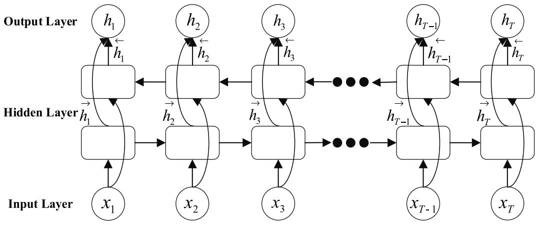
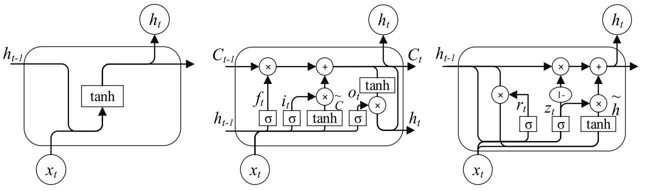


FIGURE 2. The architecture of a single-layer bidirectional RNN.

At present, there are countless learning targets that require dealing with sequential data, especially in tasks like machine translation and speech recognition. Progress in this area owes to advances in model architectures, training algorithms, and parallel computing [23]. Unfortunately, in practice, the basic RNN model does not handle long sequences well. One of the main reasons for this is that gradient explosion and gradient vanishing occur frequently during the training process. This causes the training gradient to not be passed on in long sequences, which prevents the RNN from capturing long-distance effects.



(a) RNN unit **(b) LSTM unit** **(c) GRU unit**

FIGURE 3. The structure of RNN, LSTM and GRU units.

The LSTM model shown in **Figure 3(b)** is a type of improved RNN, capable of learning long-term dependencies [24]. It is more complex in structure and consists of a series of gate units but is easier to train since it effectively avoids gradient explosion and vanishing gradients [25]. In summary, LSTM has successfully solved the defects of the original RNNs. It became the most popular RNN unit for a time, and has been widely applied in various fields.

However, the complexity of the LSTM model brings with it considerable computational cost. Of the various LSTM variants, the GRU unit may be the most successful [26]. As shown in **Figure 3(c)**, it significantly simplifies the LSTM without affecting the result. Specifically, GRU makes two major changes to the LSTM. First, it transforms the input gate (i_t), forget gate (f_t), and output gate (o_t) into an update gate (z_t) and a reset gate (r_t). Second, it combines unit states and outputs into one state ($\tilde{h}_t$), which is defined by:

$$\begin{aligned} z_t &= \sigma\left(W_z \cdot [h_{t-1}, x_t] + b_z\right) \\ r_t &= \sigma\left(W_r \cdot [h_{t-1}, x_t] + b_r\right) \\ \tilde{h}_t &= tanh\left(W_h \cdot [r_t * h_{t-1}, x_t] + b_h\right) \\ h_t &= z_t * \tilde{h}_t + (1 - z_t) * h_{t-1} \end{aligned} \tag{2}$$

where $*$ denotes the element-wise product.

In some cases, the output of the current moment is related not only to the previous states but also to future states. Thus, bidirectional RNN [27], which was designed for such cases, accomplishes the task by simultaneously training the model in both the forward and backward time directions. Accordingly, this paper deploys a Bi-GRU network as the core architecture.

B. ATTENTION MECHANISM

The encoder-decoder framework was proposed in [26]. It consists of two RNN models that can be different units (e.g., basic RNN, LSTM or GRU). As shown in **Figure 4**, the encoder is an RNN model utilizing a certain unit that inputs each sampling point x sequentially. As the samples are fed into the model by time step, the hidden state of the encoder is a summary vector c_t of the entire input signal, and the hidden state of each time step is computed by Eq. (1). The decoder is another RNN model trained to generate the output by predicting the next status $y_{t'}$ given the summary vector c_t. Overall, the predicting probability of the decoder is defined through the ordered conditionals:

$$p\left(y\right) = \prod_{t'=1}^{T} p\left(\left\{y_{t'} \mid y_1, y_2, \cdots, y_{t'-1}\right\}, c_t\right) \tag{3}$$

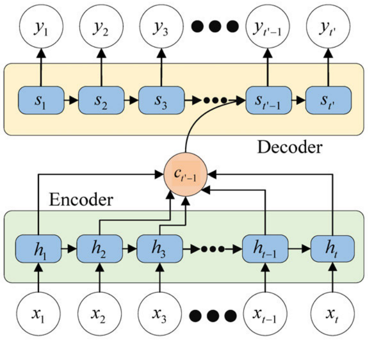


FIGURE 4. Introducing attention mechanism to the encoder-decoder framework.

Here, each conditioned probability is defined as:

$$p\left(\left\{y_{t'} \mid y_1, y_2, \cdots, y_{t'-1}\right\}, c_t\right) = u\left(y_{t'-1}, s_{t'}, c_t\right) \tag{4}$$

where u is also a nonlinear function to generate the probability of y_t, and it can be multiple forms such as a multi-layer model. Obviously, $y_{t'}$ are determined by $y_{t'-1}$, $s_{t'}$ as well as c_t. Thus, the hidden state $s_{t'}$ of the decoder at time step t' is computed by:

$$\mathrm{s}_{t'} = f\left(\mathrm{s}_{t'-1}, \mathrm{y}_{t'-1}, c_t\right) \tag{5}$$

To explore the importance of sampling points for predicting an output in each time step, an attention-based mechanism was proposed in [28], as shown in **Figure 4**. The summary vector $c_{t'}$ is computed, after which a weighted sum of the hidden state h_t in the encoder is obtained:

$$c_{t'} = \sum_{t=1}^{T_x} \alpha_{t't} h_t \tag{6}$$

where the $\alpha_{t't}$ denotes the importance mentioned above, which is the output of time step t' corresponding to input h_t and is abstractly computed as:

$$\alpha_{t't} = \frac{exp\left(e_{t't}\right)}{\sum_{t=1}^{T_x} exp\left(e_{t't}\right)} \tag{7}$$

where

$$e_{t't} = a\left(s_{t'-1}, h_t\right)$$

is a scoring method that measures the degree of matching between the inputs around time step t of the encoder and the output at time step t'.

After introducing the attention mechanism, what we want is that the context used in predicting the output at each moment is the context that is related to the current output. In essence, it uses the hidden state to enhance the ability to selectively memorize the sampling values in newly generated sequence segments.

III. ATTENTION-BASED EQUITABLE SEGENTATION GATED RECURRENT UNIT NETWORKS

Based on the theoretical derivation in the previous section, a novel framework termed the Multi-sensory Attention-based Equitable Segmentation Gated Recurrent Unit Network is proposed, whose overall architecture is shown in **Figure 5**. It mainly consists of three parts: data temporal correlation pre-processing (i.e., equitable segmentation) module, a single-layer bidirectional GRU (i.e., the encoder) model, and a sampling point-level attention layer.

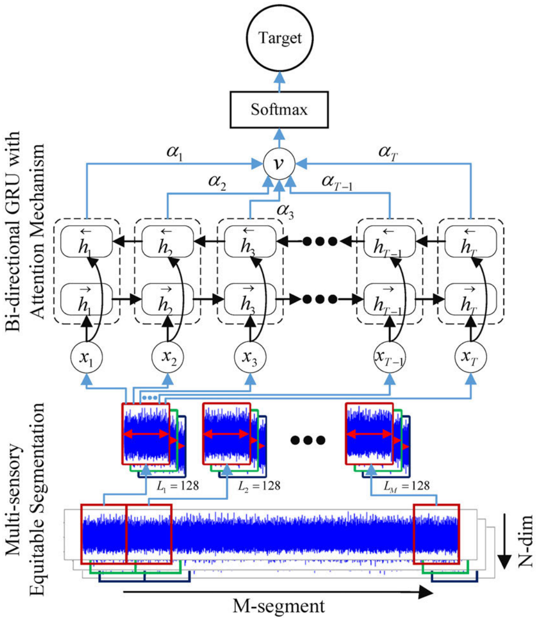


FIGURE 5. The architecture of AESGRU.

A. MULTI-SENSORY EQUITABLE SEGMENTATION

In different industrial scenarios, the type of the collected data and the number of deployed sensors may not be always identical. Besides, the sampling rate is generally extremely high, which may cause the entire sensor sequence to be extended indefinitely. Assuming that the raw sensor signal is N-dimensional, since the sampling frequency and temporal characteristics are consistent, the multi-dimensional signals can be aligned. Traditional feed-forward neural networks treat sensor sampling points as independent values. But it does not take into account the inherent temporal correlation of the sensor signal. However, the unique memory capabilities of the GRU network make the model specialize in the learning and representation of time-series. Therefore, we divide the entire sensor sequence into M segments with equal length, and each segment is a relatively short sub-time series that can be denoted as:

$$x = \begin{bmatrix} x_{11}, & x_{12}, & \cdots, & x_{1n} \\ x_{21}, & x_{22}, & \cdots, & x_{2n} \\ \vdots, & \vdots, & \ddots, & \vdots \\ x_{m1}, & x_{m2}, & \cdots, & x_{mn} \end{bmatrix}$$

where the x denotes the actual value of the raw sensor signal at each sampling point.

Each sub-segment consists of a certain number of sensor values, which has preserved the temporal correlation of the complete sequence. Besides, these newly generated samples are provided with the characteristic of independent and identical distribution. One sub-segment will be taken as one training sample for our deep model, and it fits well with the input requirements of the GRU network. The total number of training samples is equal to the number of sub-segments (i.e., M). Notably, each sub-segment is of fixed-length L (e.g., L = 64, 128 or 256, etc.), which is a hyper-parameter corresponding to the time step of GRU network. That is, the input time step of the GRU model should be set to L. If the sensor sequence is segmented from the sampling starting point, the number of data points at the final segment may be less than L (in most cases). At this point, the segment data may be skipped or spliced with the subsequent sequence if it is compatible. Finally, these preprocessed raw signals are fed directly into the model. In addition, our proposed model not only supports multi-sensory input, but also eliminates complicated feature engineering and completely implements an end-to-end health perceiving system.

B. BIDIRECTIONAL GRU WITH ATTENTION MECHANISM

As depicted in **Figure 5**, assuming that the length of each segment is set to T (e.g., T = 256) and the sampling point in each segment can be expressed as x_t, $t \in [1, T]$, the RNN tends to better represent recent inputs, with the hidden state h_t focusing on sampling points around x_t. Moreover, the performance of industrial equipment will gradually degrade after it a stable stage, and the transformation of sensor values is of short-term relevance. Thus, we design a bidirectional GRU to obtain annotations of sampling points by summarizing information from both directions, which consists of the forward GRU $\vec{h}_t$ and the backward GRU $\overleftarrow{h}_t$. The former learns the raw sub-sequence from x_1 to x_T and computes the forward hidden states $(\vec{h}_1, \vec{h}_2, \cdots, \vec{h}_t)$. The latter learns in the reverse order (i.e., from x_T to x_1) and generates the backward hidden states $(\overleftarrow{h}_1, \overleftarrow{h}_2, \cdots, \overleftarrow{h}_t)$. These are denoted as follows:

$$\begin{aligned} \vec{h}_t &= \overrightarrow{GRU}\left(x_t\right), \quad t \in [1, T], \\ \overleftarrow{h}_t &= \overleftarrow{GRU}\left(x_t\right), \quad t \in [T, 1]. \end{aligned} \tag{8}$$

Notably, the T indicates the length of time step of GRU network, at the same time refers to the length of each sub-segment, thus it has the same meaning with variable L mentioned in the Section III-A.

We concatenate the forward hidden state $\vec{h}_t$ and the backward hidden state $\overleftarrow{h}_t$ to obtain an annotation of the health condition h_t, i.e., $h_t = [\vec{h}_t, \overleftarrow{h}_t]$. In this way, the comprehensive hidden state h_t contains both the preceding sampling points and the subsequent sampling points around x_t.

Because there are only one target need to generate, therefore in our model architecture, we define the conditional probability in Eq. (3) as:

$$p\left(y \mid x\right) = f\left(v\right) \tag{9}$$

Here, we use the f function, a softmax classifier, to obtain the final predicting result. The v denotes the summary vector that is learned by the Bi-GRU, which depends on a sequence of hidden state $(h_1, h_2, \cdots, h_T)$ represented by raw input x.

Obviously, we have treated these segmented sensor sequences as time-series data, and have learned the representation of equipment status in both the forward and backward directions. Despite all that, it is noteworthy that not all sampling point values contribute equally to the representation of the health status. To select and reward those sampling points that correctly diagnose faults, we introduce an attention mechanism [29] to extract raw sampling points that indeed have a significant impact on the health of industrial equipment, and combine the representation of those data points to form the final health perception vector v. First, we feed the hidden states h_t into a one-layer fully connected neural network to acquire a hidden representation d_t. Second, we introduce a context vector d_s and measure the significance of the sampling points via the vector. The vector is computed as follows:

$$d_t = tanh\left(W_s h_t + b_s\right) \tag{10}$$

$$\alpha_t = \frac{exp\left(d_t^T d_s\right)}{\sum_{t=1}^{T} exp\left(d_t^T d_s\right)} \tag{11}$$

The context vector v is computed as a weighted sum of these hidden states h_t:

$$v = \sum_{i=1}^{T} \alpha_t h_t \tag{12}$$

The vector v, finally, can be used as the feature vector for fault diagnosis by means of a softmax classifier.

$$h_\theta\left(x_t\right) = softmax\left(v\right) \tag{13}$$

Here, we deploy a "many to one" model architecture that corresponds to the encoder-decoder framework mentioned in Section II-B. The encoder refers to the Bi-GRU network, and the decoder refers to the softmax classifier in our model. Therefore, the dimension of the vector v is just 1×1, and the context vector d_s can be considered an abstract representation for selecting the informative sampling points, which are randomly initialized and learned during the training process.

C. SOFTMAX CLASSIFIER AND COST FUNCTION

Suppose we have a sample set $\{x^{(i)}, y^{(i)}\}_{i=1}^{M}$, which consists of multiple inputs and their labels, where $x^{(i)} \epsilon \mathbb{R}^N$ and $y^{(i)} \epsilon \{1, 2, \cdots, \mathrm{K}\}$. For each input sample, the softmax classifier will calculate the probability of the sample for each label. Consequently, it will output a vector that contains K elements, in which each value indicates the probability of the sample belonging to a specific label. In our study, we perform a four-class diagnostic task whose expression is as follows:

$$h_\theta\left(x_t\right) = \begin{bmatrix} p\left(y_t = 0 | x_t; \theta\right) \\ p\left(y_t = 1 | x_t; \theta\right) \\ p\left(y_t = 2 | x_t; \theta\right) \\ p\left(y_t = 3 | x_t; \theta\right) \end{bmatrix} = \frac{1}{\sum_{j=0}^{3} e^{\theta_j^T x_t}} \begin{bmatrix} e^{\theta_0^T x_t} \\ e^{\theta_1^T x_t} \\ e^{\theta_2^T x_t} \\ e^{\theta_3^T x_t} \end{bmatrix} \tag{14}$$

Subsequently, our model is trained by minimizing the cost function, which is defined as follows:

$$\begin{aligned} J\left(\theta\right) = -\frac{1}{m}\left[\sum_{i=1}^{m}\sum_{j=0}^{3} 1\left\{y^{(i)} = j\right\} log \frac{e^{\theta_j^T x_t}}{\sum_{j=0}^{3} e^{\theta_j^T x_t}}\right] \\ + \frac{\lambda}{2}\sum_{i=1}^{m}\sum_{l=1}^{L}\theta_{il}^2 \end{aligned} \tag{15}$$

Obviously, our model can handle m samples at a time and consists of two parts. The left part represents the model prediction loss, which is used to measure the degree of fitting between the model and sample, where $1\{\cdot\}$ is an indicative function that returns 1 if the value in the parentheses is true and 0 otherwise. The right part is a regularization term used to modify the cost function. This decay term will punish excessively large parameters by tuning the hyper-parameter λ, whose value is greater than 0. L denotes the length of each segment (e.g., 256), corresponding to the time step of Bi-GRU network.

IV. EXPERIMENTS

This section will detail the modeling process. Besides, we will evaluate our proposed model on the CWRU dataset.

A. DESCRIPTIONS OF DATASET

The experimental data used here were provided by the bearing data center at CWRU [30]. The vibration signal was collected

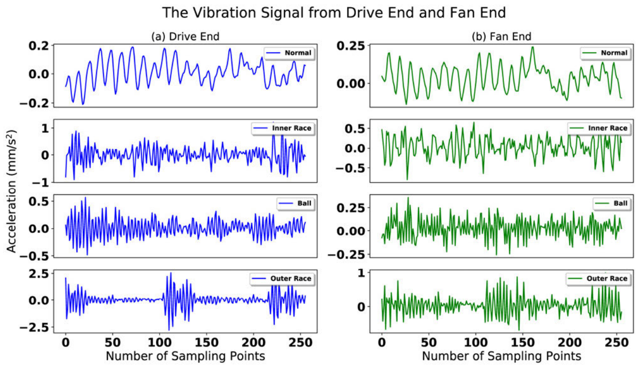


FIGURE 6. Raw vibration signal sequence of Reliance Electric Motor.

from a motor test rig that contained single-point defects at the drive end and fan end, as well as from normal operating conditions. **Figure 6** shows the signal sequence at the drive end and fan end under four different conditions. The sampling time at both ends is consistent and the sampling frequency is identical. Therefore, this paper uses the 0.007'' fault diameter data with 12 *kHz* sampling.

These vibration signals constitute the motor bearing dataset, and the collected data at 1797 rpm (0 *hp*) is utilized for our study. In this experiment, we require complete a four-class diagnostic task. Unlike [14]–[16], considering that the vibration sensor signal is typical time-series data with temporal characteristics, we adopt RNN models and its variants, including LSTM and GRU units, which are good at handling such data. Therefore, different from [21], our work utilizes the vibration signal acquired at both the drive end and fan end. The data are extended to 2-D matrix and is directly fed into our model to learn the fault representation.

B. EXPERIMENTAL SETUP

In this section, we will evaluate the effectiveness of the λ proposed method through experiments. The core task is to verify whether better results can be obtained in the fault diagnosis of industrial equipment when the attention mechanism is integrated into the Bi-GRU model to handle multi-dimensional sensor sequences. We have designed three experiments to verify this hypothesis.

To evaluate the performance of our proposed AESGRU, we perform extensive experiments on the CWRU dataset. First, to enable the model with effectiveness, the original dataset is divided into an 8:2 ratio of training data to test data through a random sampling method. In specific, there are 80% of data extracted from the entire dataset to be taken as the training set, and the rest data will be used for testing. As described in the previous sections, we concentrated on implementing a complete end-to-end diagnostic model. Therefore, we directly input the raw data (i.e., the acceleration value of the vibration signal) into the model for learning representation. Then, we choose accuracy, training time and prediction time as performance indicators to measure the merits of the model, after which a confusion matrix is utilized to observe the distribution of the test results. We utilized the grid search method to find the optimal parameters during the training process. The tuning options for hyper-parameters in model training are listed in **Table 1**. Our AESGRU reached the highest accuracy when the learning rate was 0.0025, the regularization rate was 0.0015, and the time step size as set to 128. Finally, we choose basic RNN, LSTM, GRU, and Bi-GRU structures for training and fault prediction, respectively, to further analyze and compare their performance metrics.

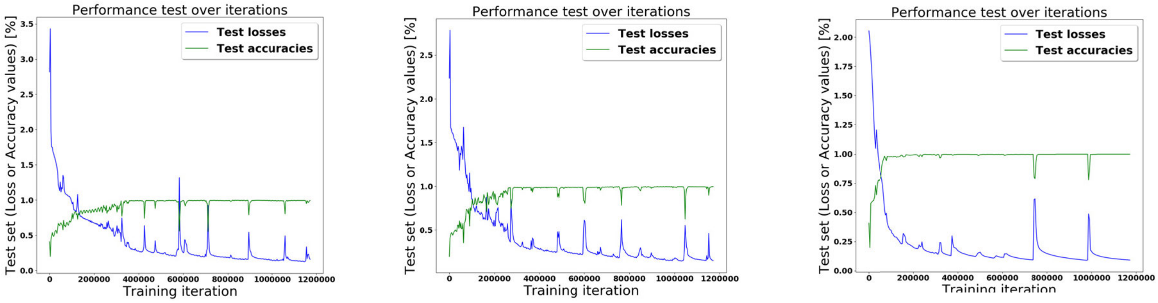
TABLE 1. The hyper-parameters selection range of model training process.

Hyper-parameters	Range
Learning rate	[0.0005, 0.001 , 0.0025, 0.005, 0.01]
λ (L2 regularization term)	[0.0001, 0.0005, 0.001, 0.0015, 0.005]
Batch-size	[50, 100, 150, 200, 250, 300, 500]
Time step	[32, 64, 128, 256, 512, 1024]
Epoch (Training)	[50, 100, 200, 300, 500, 1000]

In our work, we deal with 2-D raw sensor sequences and directly cut the entire sequence proportionally with the time step interval. However, the required length of each segment to better represent the operating status as well as to obtain a better temporal correlation with sequence segments is also investigated. Here, the length of each segment corresponds to the time step of our proposed model. To analyze and verify the temporal characteristics of this end-to-end system at the input end, we set up the second experiment to compare the AESGRU model horizontally. This verification is implemented through segmenting the entire sensor sequence at sizes of [32, 64, 128, 256, 512] without changing other hyper-parameter settings, and set the time step of our model to the corresponding sizes.

The input sensor sequence consists of discrete values of equal length which are fed into the model in chronological order to generate an output at each time step. The proposed model does not directly use these outputs as the final result, but introduces an attention mechanism that adds the weights for the output of each time step. The weighted sum is considered the final prediction result. These weights reflect the impact of each sample value of the sensor sequence with regard to fault diagnosis. Finally, the attention distribution is visualized by weight values curves.

To summarize, compared to traditional learning methods, the use of RNN with GRU cells requires little to no feature engineering. Data can be fed directly into the model, which acts like a black box. However, other studies on the use of PdM with industrial equipment utilize a great deal of feature engineering. The drawbacks of these approaches have been described in the previous sections. As explained in [21], an RNN takes a batch of input vectors to learn the representation and output other vectors. In our experiments, a ''many to one'' architecture is utilized: we transform the time-series representation of feature vectors (one vector per time step)



(a) GRU model of single-layer (b) Bi-GRU model of single -layer (c) Our AESGRU model

FIGURE 7. The test accuracies and losses during training process for three different GRU architecture on CWRU. The green curves indicate the accuracy of these three architectures, and the blue curves indicate the loss of these three architectures over training iterations.

and output a probability vector for fault classification. These feature vectors are extracted from raw vibration signals. Due to the small sample size of the training dataset, the model is prone to overfitting. To address this, we utilize L2 regularization. All of the experimental models were trained using the Adam [31] optimizer with a learning rate of 0.0025 and a minibatch size of 150.

The aforementioned experiments were conducted on a 3.20 GHz Intel CPU.

C. EXPERIMENTAL RESULTS

The performance metrics are listed in **Table 2**, including those for four selected algorithms and our proposed AESGRU method. The test accuracy and loss of the GRU, Bi-GRU, and AESGRU during the training process are shown in **Figure 7**. First, we can observe that the RNN models all achieve excellent performance on the sensor sequence when dealing with fault diagnosis of roller bearings. Several different unit structures and model architectures were especially helpful for improving the overall performance. We set the time step to 128 and used a single-layer model architecture. Subsequently, these models were trained for 300 epochs and the best test loss was used as the final result. Second, it can be observed intuitively that the basic RNN with tanh activation function is less accurate and takes longer to train. It exhibits poor stability at the initial training stage, and the convergence rate is slow because the basic RNN is prone to vanishing gradients. The LSTM and GRU units greatly improve training performance, and our proposed method is superior to other models. It can be shown that RNN models are suitable for fault diagnosis and prediction tasks.

TABLE 2. Metrics for different RNN architectures. Besides using a learning rate of 0.0001 and an iteration number of 1500 to train the RNN model, other parameters are identical to those of the LSTM and GRU model.

RNN Unit Types	No. Model Layer	Accuracy	Training Time	Testing Time
Basic RNN	single-layer	98.81%	1018.85 s	**0.037 s**
LSTM		99.41%	666.7 s	0.083 s
GRU		99.64%	779.9 s	0.105 s
Bi-GRU		99.76%	**636.3 s**	0.101 s
AESGRU		**99.88%**	1183.2 s	0.152 s

In the second experiment, we used different step sizes to equally divide the dataset, and set the time step of the model to the same length. We attempted to find the best segmentation step size, and the performance metrics are listed in **Table 3**. It can be observed that the number of samples obtained after equal segmentation of the dataset is different. Obviously, as the segmentation step becomes shorter, the number of samples will increase. On the contrary, the number of samples will be reduced (and vice versa). More specifically, when L is set to 32, the sample scale is at a maximum, but the accuracy is not maximized. This may be caused by sample segmentation—that is, the model cannot learn a good representation of the machine health from 32 consecutive sampling points. It is worth noting that 100% accuracy is attained when L is set to 256.

TABLE 3. The effect of different step size on the performance of AESGRU.

The proposed method	L	Train samples/ Test samples	Test accuracy	Training time	Testing time
AESGRU	32	15620/3364	98.09%	1276.4s	0.245s
	64	7810/1682	99.66%	1281.4s	0.254s
	128	3905/841	99.88%	1183.2s	0.152s
	256	1950/421	100%	1314.6s	0.296s
	512	975/212	98.64%	1542.5s	0.315s

It can also be shown that performing an equal segmentation with a step size of 256 significantly optimizes the performance of the model. Although the prediction accuracy is suboptimal when L is 128, the overall training time and the single prediction time are minimized. Therefore, the segmentation step should be set to 128 if efficiency is the priority. Ultimately, the test results of these five segmentation methods in the training process are plotted in **Figure 8**, and the confusion matrix is listed in **Table 4**. As we can see, the model correctly predicted all 521 test samples when L was 256, and the performance on the CWRU dataset was always 100% over the course of multiple experiments. When L was 32, 64 or 128, the model output false positives (marked with red)

TABLE 4. Test Confusion Matrix of AESGRU for different step size, including 32, 64, 128, 256, and 512. The letters "0", "1", "2", and "3" indicate four kinds of equipment degraded conditions which contains Normal, Inner race faulty, Ball faulty, and Outer race faulty. In each sub-table, the green-filled grid indicates that the classification is correct, and the red-filled grid indicates that the normal status have been predicted to faulty classes. The blue-filled grid indicates the opposite situation.

(a) The segment step size is set to 32

Predict / True		Predicted labels			
		0	1	2	3
True Labels	0	1372	0	0	0
	1	0	652	7	5
	2	0	0	630	34
	3	0	0	18	646

(b) The segment step size is set to 64

Predict / True		Predicted labels			
		0	1	2	3
True Labels	0	686	0	0	0
	1	0	331	0	1
	2	0	0	330	2
	3	0	0	3	329

(c) The segment step size is set to 128

Predict / True		Predicted labels			
		0	1	2	3
True Labels	0	343	0	0	0
	1	0	166	0	0
	2	0	0	166	0
	3	0	1	1	164

(d) The segment step size is set to 256

Predict / True		Predicted labels			
		0	1	2	3
True Labels	0	172	0	0	0
	1	0	83	0	0
	2	0	0	83	0
	3	0	0	0	83

(e) The segment step size is set to 512

Predict / True		Predicted labels			
		0	1	2	3
True Labels	0	86	0	0	0
	1	1	41	0	0
	2	3	1	38	0
	3	0	0	0	42

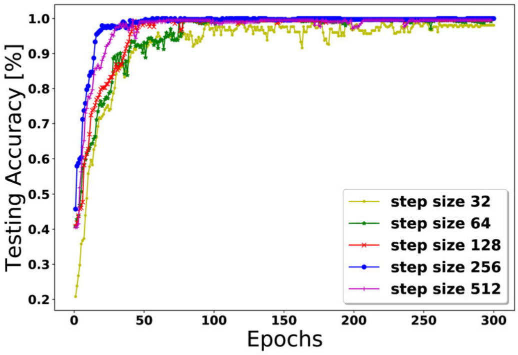


FIGURE 8. The accuracy of training epochs at the different segmentation step sizes.

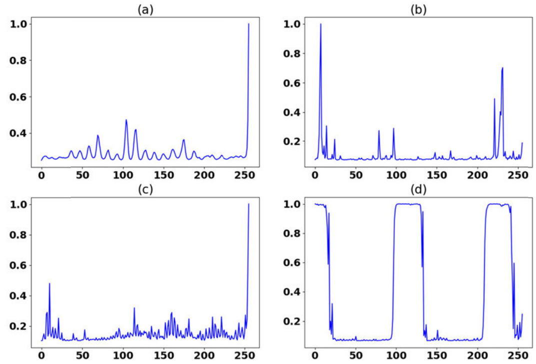


FIGURE 9. The distribution of attention weight under four equipment health conditions. The (a), (b), (c) and (d) denote the attention weight distribution of a random sensor sample in the four machine status, respectively.

in some cases. To some extent, this is acceptable, since no irreversible damage was caused. In other words, this simply adds a nominal amount to maintenance costs, which is minimal compared to expensive downtime. However, when L is 512, the model performs particularly poorly and outputs false negatives (marked with blue). This is the most dangerous result in a fault prediction task because it presents a security risk to the industrial system (e.g., system paralysis). Therefore, the prediction accuracy should be as close as possible to 100% to reflect the significance of data-driven PdM. This is the fundamental reason why RNN models can achieve better performance in fault diagnosis. However, eliminating the loss caused by these false predictions will be discussed in future work.

After equal segmentation of the raw signal, each segment contains an identical number of sampling points. Strictly speaking, these sampling points are essentially a time series, and the specific values of the previous sampling points can not only represent the equipment operation status at the corresponding moment but also have a significance effect on subsequent series. Accordingly, the sensor sequence is context dependent. To verify that our model can capture the importance of sampling points with regard to context, we plot the distribution of the attention weights for four types of equipment status, shown in **Figure 9(a-d)**, respectively. Certainly, this is only the α distribution from one set of samples, but our model can build an α distribution scheme for any sample. We can observe that all of the segments have a corresponding attention weight distribution, which will be assigned to each point and has a weight range from 0 to 1. This indicates that our model can capture diverse context and assign context-dependent weights to the sampling points.

To confirm that our model can select informative sampling points in each segmentation, we visualize the distribution of attention weights corresponding to the raw sensor sequence in **Figure 10**. For convenience, we have normalized the data, so each red bars in the sub-figures tend to 1. The left y-axis represents the acceleration values from the two end bearings, and the right y-axis denotes the α values. **Figure 10** shows that our model can select sampling points that carry fault information. For example, for the sample with label 3 (**Fig. 10(d)**), the vibration of the raw signal is particularly severe in some places (marked yellow), and the maximum acceleration is closer to 3 mm/s^2. Our model can concentrate

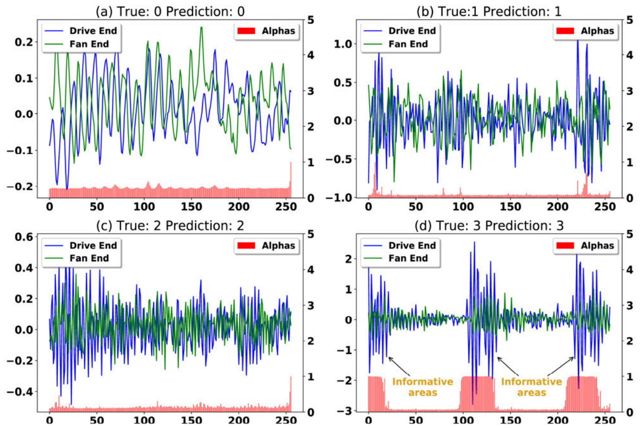


FIGURE 10. The distribution of attention weights corresponding to the raw sensor sequence. (a), (b), (c), and (d) indicate the relationship of four samples with the corresponding attention weight. The blue curve denotes the sampling points at the drive end, and the red curve denotes the sampling points at the fan end. The red bars denote the distribution of the attention weights.

on these areas and assign a higher weight to each sampling point, which is more informative for fault diagnosis. That is, when the model focuses on learning how to represent equipment fault types, it will pay more attention to this part of the data. The same is true for label 0, label 1, and label 2.

D. COMPARISON WITH OTHER WORKS

The sample capacity of the CWRU dataset is somewhat insufficient, a characteristic that is mitigated by the design method of our AESGRU model (i.e., equitable segmentation of the entire dataset). Furthermore, the time complexity is critical for real-time prediction systems. The diagnostic accuracy and prediction time consumption using different segmentation step sizes are shown in **Figure 11**. Accordingly, taking into account the number of test samples and time taken, an accuracy of 99.88% is considered the final result of our model (with L set to 128).

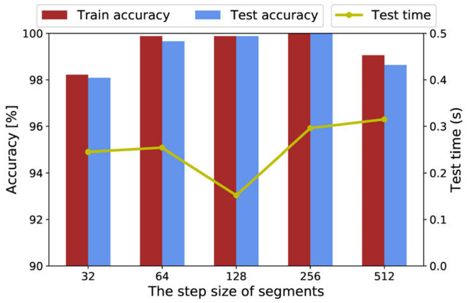


FIGURE 11. The diagnostic accuracy and test time consumption of AESGRU under different segmentation step size.

To demonstrate the effectiveness of our proposed approach, the diagnostic accuracy of several related works were compared using the same CWRU dataset. The details of this comparison are presented in **Table 5**. Specifically, Zhang *et al.* [32] proposed the ICDSVM model, which was optimized by inter-cluster distance (ICD) in the feature space and achieved 97.75% accuracy but also required manual feature extraction. Zhang *et al.* [33] deployed a four-layer DNN and achieved 94.4% diagnostic accuracy. Due to unlabeled data being easier to acquire than labeled data, Lei *et al.* [14] and Li *et al.* [16] directly learned features from mechanical signals via unsupervised learning, attaining accuracies of 99.66% and 97.29%, respectively. In addition, Zhao *et al.* [21] combined handcrafted feature design with automatic feature learning, and achieved 99.6% accuracy. In particular, Li *et al.* [34] integrated a two-layer CNNs with a Bi-LSTM network, at the same time the attention mechanism is introduced behind the LSTM units, finally achieved 99.74% accuracy on the CWRU dataset. Compared with these existing works, our AESGRU method has achieved optimal results from the perspective of fault prediction accuracy. Above all, the new sample set is made into time-series segments with temporal correlation through the equitable segmentation of the original dataset. Thus, the model enables end-to-end learning from strongly correlated data, which is integrated with multi-dimensional sensor signal from the vibration data of drive end and fan end. On top of that, the attention mechanism allows our AESGRU to selectively screen out a small number of important sampling points from long sequence segments and focus limited attention resources on these points, so that greatly improving the efficiency and accuracy of health perception.

TABLE 5. Comparison of test accuracy with other research works on CWRU dataset.

Method	Accuracy
ICDSVM [32]	97.75%
Sparse Filtering [14]	99.66%
Auto-Encoder [16]	97.29%
DNN with 4-layers [33]	94.40%
LFGRU (Bi-GRU) [21]	99.60%
CNN+LSTM+Attention [34]	99.74%
The proposed AESGRU	99.88%

In general, we have further improved the diagnostic accuracy of health perception by using equitable segmentation of original dataset and introducing attention mechanism. Although the improvement is especially tiny weak, we believe that any effort to improve the robustness of diagnostic system is worthwhile. It should be pointed out, however, that a single metric alone cannot judge the absolute performance of a model. In some cases, we should comprehensively consider some factors, such as model complexity, time overhead, etc., to judge the pros and cons of a deep model.

V. CONCLUSION

In this work, a novel diagnostic model, named AESGRU, is proposed for the health perception of rotating machinery. After equitable segmentation of the raw sensor sequence, the newly generated samples with temporal correlation are fed directly into this model, so that it enables end-to-end learning from these strongly correlated data. In particular, the attention mechanism is effective in modeling the long-term dependency of sensor sequences. Besides, it not only eliminates the need for feature engineering, but also is applicable to multi-sensor scenarios. The experimental results demonstrate that when the segmentation step size is set to 256, the accuracy of 100% can be attained.

It has been verified that the attention mechanism can make health perception of massive sensor sequences more accurate. In future work, we will continue to focus on attention between sequences. Meanwhile, our proposed method will be utilized for actual industrial equipment, specifically to *washing automatic equipment*, which is an automatic device used to clean high-speed rails and subways before repairs.

REFERENCES

[1] W. Zhang, D. Yang, and H. Wang, "Data-driven methods for predictive maintenance of industrial equipment: A survey," *IEEE Syst. J.*, vol. 13, no. 3, pp. 2213–2227, Sep. 2019.

[2] Q. Yang, C. Hu, and N. Zheng, "Data-driven diagnosis of nonlinearly mixed mechanical faults in wind turbine gearbox," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 466–467, Feb. 2018.

[3] T. de Bruin, K. Verbert, and R. Babuska, "Railway track circuit fault diagnosis using recurrent neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 3, pp. 523–533, Mar. 2017.

[4] Y. Xu, Y. Sun, J. Wan, X. Liu, and Z. Song, "Industrial big data for fault diagnosis: Taxonomy, review, and applications," *IEEE Access*, vol. 5, pp. 17368–17380, 2017.

[5] J. Lee, F. J. Wu, W. Y. Zhao, M. Ghaffari, L. X. Liao, and D. Siegel, "Prognostics and health management design for rotary machinery systems-reviews, methodology and applications," *Mech. Syst. Signal Process.*, vol. 42, nos. 1–2, pp. 314–334, Jan. 2014.

[6] C. A. Tokognon, B. Gao, G. Y. Tian, and Y. Yan, "Structural health monitoring framework based on Internet of Things: A survey," *IEEE Internet Things J.*, vol. 4, no. 3, pp. 619–635, Jun. 2017.

[7] G. A. Susto, A. Schirru, S. Pampuri, S. McLoone, and A. Beghi, "Machine learning for predictive maintenance: A multiple classifier approach," *IEEE Trans. Ind. Informat.*, vol. 11, no. 3, pp. 812–820, Jun. 2015.

[8] J. J. A. Costello, G. M. West, and S. D. J. McArthur, "Machine learning model for event-based prognostics in gas circulator condition monitoring," *IEEE Trans. Rel.*, vol. 66, no. 4, pp. 1048–1057, Dec. 2017.

[9] M. Canizo, E. Onieva, and A. Conde, S. Charramendieta, and S. Trujillo, "Real-time predictive maintenance for wind turbines using Big Data frameworks," in *Proc. IEEE Int. Conf. Prognostics Health Manage.*, Jun. 2017, pp. 70–77.

[10] S. Hong and Z. Zhou, "Application of Gaussian process regression for bearing degradation assessment," in *Proc. IEEE Int. Conf. New Trends Inf. Sci. Service Sci. Data Mining*, vol. 42, Oct. 2012, pp. 644–648.

[11] D. Ganga and V. Ramachandran, "IoT-based vibration analytics of electrical machines," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4538–4549, Dec. 2018.

[12] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015.

[13] M. Z. Alom, T. M. Taha, S. Westberg, P. Sidike, M. S. Nasrin, B. C. Van Esesn, A. A. S. Awwal, V. K. Asari, and C. Yakopcic, "The history began from AlexNet: A comprehensive survey on deep learning approaches," 2018, *arXiv:1803.01164*. [Online]. Available: https://arxiv.org/abs/1803.01164#

[14] Y. Lei, F. Jia, J. Lin, S. Xing, and S. X. Ding, "An intelligent fault diagnosis method using unsupervised feature learning towards mechanical big data," *IEEE Trans. Ind. Electron.*, vol. 63, no. 5, pp. 3137–3147, May 2016.

[15] F. Jia, Y. G. Lei, J. Lin, X. Zhou, and N. Lu, "Deep neural networks: A promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data," *Mech. Syst. Signal Process.*, vols. 72–73, pp. 303–315, May 2016.

[16] C. Li, W. Zhang, G. Peng, and S. Liu, "Bearing fault diagnosis using fully-connected winner-take-all autoencoder," *IEEE Access*, vol. 6, pp. 6103–6115, 2017.

[17] R. Zhao, J. Wang, K. Mao, and R. Yan, "Machine health monitoring with LSTM networks," in *Proc. IEEE Int. Conf. Sens. Technol.*, Nov. 2016, pp. 1–6.

[18] Y. Mei, Y. Wu, and L. Lin, "Fault diagnosis and remaining useful life estimation of aero engine using LSTM neural network," in *Proc. IEEE Int. Conf. Aircraft Utility Syst.*, Oct. 2016, pp. 135–140.

[19] R. Zhao, R. Yan, J. Wang, and K. Mao, "Learning to monitor machine health with convolutional Bi-directional LSTM networks," *Sensors*, vol. 17, no. 2, p. 273, 2017.

[20] P. Malhotra, V. Tv, A. Ramakrishnan, L. Vig, P. Agarwal, G. Shroff, and G. Anand, "Multi-sensor prognostics using an unsupervised health index based on LSTM encoder-decoder," 2016, *arXiv:1608.06154*. [Online]. Available: https://arxiv.org/abs/1608.06154#

[21] R. Zhao, D. Wang, K. Mao, F. Shen, J. Wang, and R. Yan, "Machine health monitoring using local feature-based gated recurrent unit networks," *IEEE Trans. Ind. Electron.*, vol. 65, no. 2, pp. 1539–1548, Feb. 2018.

[22] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, vol. 1. Cambridge, MA, USA: MIT Press, 2016.

[23] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A critical review of recurrent neural networks for sequence learning," 2015, *arXiv:1506.00019*. [Online]. Available: https://arxiv.org/abs/1506.00019#

[24] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[25] T. H. Trinh, A. M. Dai, Q. V. Le, and M.-T. Luong, "Learning longer-term dependencies in RNNs with auxiliary losses," 2018, *arXiv:1803.00144*. [Online]. Available: https://arxiv.org/abs/1803.00144#

[26] K. Cho, B. Van Merrienboer, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, and C. Gulcehre, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," 2014, *arXiv:1406.1078*. [Online]. Available: https://arxiv.org/abs/1406.1078#

[27] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.

[28] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, *arXiv:1409.0473*. [Online]. Available: https://arxiv.org/abs/1409.0473

[29] Z. Yang, D. Yang, X. He, A. Smola, E. Hovy, and C. Dyer, "Hierarchical attention networks for document classification," in *Proc. Conf. North Amer. Chapter Assoc. for Comput. Linguistics, Human Lang. Technol.*, 2016, pp. 1480–1489.

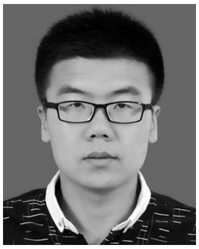
[30] *Case Western Reserve University Bearing Data Center*. Accessed: 2018. [Online]. Available: http://csegroups.case.edu/bearingdatacenter/pages/download-data-file

[31] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: https://arxiv.org/abs/1412.6980

[32] X. Zhang, Y. Liang, Y. Zang, and J. Zhou, "A novel bearing fault diagnosis model integrated permutation entropy, ensemble empirical mode decomposition and optimized SVM," *Measurement*, vol. 69, pp. 164–179, Jun. 2015.

[33] R. Zhang, Z. Peng, B. Yao, Y. Guan, and L. Wu, "Fault diagnosis from raw sensor data using deep neural networks considering temporal coherence," *Sensors*, vol. 17, no. 3, p. 549, 2017.

[34] X. Li, W. Zhang, and Q. Ding, "Understanding and improving deep learning-based rolling bearing fault diagnosis with attention mechanism," *Signal Process.*, vol. 161, pp. 136–154, Aug. 2019.



WEITING ZHANG received the M.S. degree from the Inner Mongolia University of Technology, China, in 2017. He is currently pursuing the Ph.D. degree in communication and information systems from Beijing Jiaotong University, Beijing. His specific areas of research interest mainly focus on deep learning, predictive maintenance, and wireless sensor networks.

DONG YANG (M'11) received the B.S. degree from Central South University, China, in 2003, and the Ph.D. degree in communications and information science from Beijing Jiaotong University, Beijing, China, 2009. From March 2009 to June 2010, he was a Postdoctoral Research Associate with Jonkoping University, Jonkoping, Sweden. In August 2010, he joined the School of Electronic and Information Engineering, Beijing Jiaotong University, where he is currently a Professor. His research interests include network technologies, including the Internet architecture, the industrial Internet, and wireless sensor networks.

HONGCHAO WANG (M'14) received the B.S. degree in communication engineering and the Ph.D. degree in communication and information systems from Beijing Jiaotong University, Beijing, China, in 2005 and 2012, respectively, where he is currently with the School of Electronic and Information Engineering. His research interests include the Internet architecture, network security, and wireless sensor networks.

JUN ZHANG graduated from the Changchun Institute of Technology, China, 2006, majoring in automation specialty. He joined Beijing Sheenline Group Company Ltd., in March 2007, and engaged in the development of automation equipment and automatic production line. In September 2014, he began to research factory information systems and big data application systems as well as equipment asset and health management systems.

MIKAEL GIDLUND (M'98–SM'16) received the M.Sc. and Ph.D. degrees in electrical engineering from Mid Sweden University, Sundsvall, Sweden, in 2000 and 2005, respectively.

In 2005, he was a Visiting Researcher with the Department of Informatics, University of Bergen, Bergen, Norway. From 2006 to 2007, he was a Research Engineer and Project Manager, responsible for wireless broadband communication at Acreo AB, Kista, Sweden. From 2007 to 2008, he was a Senior Specialist and Project Manager with responsibility for next-generation IP-based radio solutions at Nera Networks AS, Bergen. From 2008 to 2013, he was a Senior Principal Scientist and Global Research Area Coordinator of wireless technologies in ABB Corporate Research, with main responsibility to drive technology and strategy plans, standardization, and innovation in the wireless automation area. Since 2014, he has been a Full Professor of computer engineering with Mid Sweden University. He has pioneered the area of industrial wireless sensor network and he holds more than 20 patents (granted and pending applications) in the area of wireless communications. He has authored or coauthored more than 100 scientific publications in refereed fora. His research interests include wireless communication and networks, wireless sensor networks, access protocols, and security. He received the Best Paper Award at the IEEE International Conference on Industrial IT, in 2014. He is currently an Associate Editor of the IEEE Transactions on Industrial Informatics and the Vice-Chair of the IEEE IES Technical Committee on Cloud and Wireless Systems for Industrial Applications.