IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 3, 2019, accepted June 19, 2019, date of publication June 27, 2019, date of current version July 17, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2925468

A Neural-Network-Based Method for RUL Prediction and SOH Monitoring of Lithium-Ion Battery

JIANTAO QU[1,2], FENG LIU[1,2], YUXIANG MA[3], AND JIAMING FAN[1,2]

[1]School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China
[2]Engineering Research Center of Network Management Technology for High Speed Railway, Ministry of Education, Beijing 100044, China
[3]School of Computer and Information Engineering, Henan University, Kaifeng 475004, China

Corresponding author: Feng Liu (fliu@bjtu.edu.cn)

This work was supported in part by the National Key R&D Program of China under Grant 2016YFB1200100.

ABSTRACT The prognostic and health management (PHM) of lithium-ion batteries has received increasing attention in recent years. The remaining useful life (RUL) prediction and state of health (SOH) monitoring are two important parts in PHM of the lithium-ion battery. Nowadays, the development of signal processing technology and neural network technology introduces new data-driven methods to RUL prediction and SOH monitoring of the lithium-ion battery. This paper presents a neural-network-based method that combines long short-term memory (LSTM) network with particle swarm optimization and attention mechanism for RUL prediction and SOH monitoring of the lithium-ion battery. Before predicting RUL of the lithium-ion battery, the Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN) is utilized for the raw data denoising, which can improve the accuracy of prediction. A real-life cycle dataset of lithium-ion batteries from NASA is used to evaluate the proposed method, and the experiment results show that when compared with traditional methods, the proposed method has higher accuracy.

INDEX TERMS Lithium-ion battery, prognostic and health management (PHM), long short-term memory (LSTM), attention mechanism.

I. INTRODUCTION

Lithium-ion battery plays an important role in storing and providing energy. Because of higher energy density, lighter weight and longer charge and discharge cycle, lithium-ion batteries have already been widely used in transportation, communication and aerospace industry [1]–[4]. However, due to the complex physical and chemical changes in the use process, the performance of the lithium-ion battery will degradation or even failure, which may result in serious safety issues and major economic losses [5]–[7]. Therefore, how to realize the prognostic and health management (PHM) of the lithium-ion battery has become a heat topic and has received tremendous research efforts [8]–[10]. This is a challenge because the expected life of a lithium-ion battery varies with the environment (e.g., temperature and humidity) and intensity of use. Even batteries with the same model will have a different service life in practical use. Therefore, it is necessary to monitor the SOH of lithium-ion batteries and predict their remaining useful life.

Prognostics and health management (PHM) is expected to provide early detection of incipient faults and predict the progression of degradation in industrial components and systems [11]–[14]. PHM of lithium-ion batteries mainly includes two important aspects, i.e., SOH monitoring and RUL prediction. In fact, SOH monitoring is the basis of RUL prediction, and accuracy of SOH monitoring directly affects the accuracy of RUL prediction. SOH is defined by the related measurement parameters of lithium-ion batteries, including terminal voltage, current and capacity, etc. There are many studies on SOH monitoring and RUL prediction. These techniques can be divided into two categories including model-based [15]–[19] and data-driven [20]–[28]. The model-based methods are mainly to analyze the physical and chemical principles of the battery and establish mathematical and physical models to characterize the process of performance degradation of the lithium-ion battery. Data-driven methods have recently drawn significant attention in PHM

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

of the lithium-ion battery. Compared with other kinds of data-driven methods, the neural network, especially the deep neural network, can approach the complex nonlinear model infinitely by training multi-layer neural networks and achieve better accuracy for prediction.

This paper presents a method named PA-LSTM, which uses long short term memory network (LSTM) and attention mechanism to build the model. Furthermore, particle swarm optimization (PSO) algorithm is introduced to optimize the key parameters and pre-training the model. When monitoring the SOH of the lithium-ion battery, incremental learning is used to update the model dynamically, which can make full use of the latest data to improve the accuracy of the model. When predicting the RUL, we use CEEMDAN to denoise the raw data. Then, the RUL model is established by using PA-LSTM method. Experiments on NASA datasets show that the proposed method has higher accuracy and high practice value.

The rest of the paper is organized as follows. The related work is provided in Section II. Section III describes preliminaries, and Section IV introduces the proposed method. Section V presents and discusses the experimental results. Conclusions are drawn in Section VI.

II. RELATED WORK

A. MODEL-BASED METHODS

Ashwin at al. developed an electrochemical battery model for aging under cyclic loading condition. This model links the battery performance with electrolyte partial molar concentration. This model can construct the capacity decline process of the lithium-ion battery [15]. Mishra et al. proposed a Bayesian hierarchical model (BHM)-based prognostics approach applied to lithium-ion batteries, which can analyze and predict the discharge behavior of such batteries with variable load profiles and variable amounts of available discharge data. The BHM approach enables inferences for both individual batteries and groups of batteries [16]. Pola et al. proposed a particle-filtering-based prognostic framework that utilizes statistical characterization of using profiles to estimate the state-of-charge (SOC) and predict the discharge time of energy storage devices [17]. Mo et al. provided a method which combined the standard PF with Kalman filter to increase the accuracy of estimation, and then a particle swarm optimization algorithm was integrated to slow down the particle degradation due to particle resampling [18]. Heng et al. presented an improved unscented particle filter (UPF) algorithm based on linear optimizing combination resampling to predict the RUL of lithium-ion batteries [19]. In one aspect, the unscented Kalman filter (UKF) was used to generate a proposal distribution as an important function for particle filtering. In the other aspect, the linear optimizing combination resampling (LOCR) algorithm was used to overcome the particle diversity deficiency.

Model-based methods have achieved good results in battery health prognostics. However, the following problems still exist. The proposed model of [15] can only be applied to specific battery materials and operating conditions. Moreover, the model parameters are also determined by the physical characteristics of the battery. Method of [16] based on an equivalent circuit model. But, the implicit relationship between internal state variables of batteries can sometimes play a decisive role in the performance of batteries, it may be neglected in this model. Furthermore, it is unrealistic to consider all the complex external conditions. Methods of [17]–[19] are all based on particle filter (PF) and Kalman filter (KF). However, this kind of methods is susceptible to noise and environmental interference, and it is difficult to track the dynamic characteristics of the load.

In summary, considering the complexity of physical and chemical changes in lithium-ion batteries, as well as the noise and the diversity of environment, it is very difficult to establish an accurate model for the decline of SOH in lithium-ion batteries.

B. DATA-DRIVEN METHODS

In data-driven methods, RUL prediction and SOH monitoring of lithium-ion batteries are usually realized by analyzing historical data, which include current, voltage, capacitance and impedance, etc. Compared with the model-based methods, data-driven methods are faster, more convenient and less complex. Some experts and scholars use traditional statistical methods and mathematical optimization methods to predict the RUL of lithium-ion batteries. Zhou et al. utilized the ensemble empirical mode decomposition (EEMD) method to decompose SOH data of batteries into multiple components and then used ARIMA method to predict each component separately. Finally, the prediction results are merged to obtain the final prediction results of RUL [20]. Zhang et al. proposed a method for estimating Box-Cox transformation parameters by least squares method and Monte Carlo method. They apply this method to RUL prediction of lithium-ion batteries [21]. Xin et al. proposed an approach to analyzing the degradation of lithium-ion batteries with the sequentially observed discharging profiles. They developed a general state-space model to approximate the discharging profile of each cycle, and the expectation-maximization (EM) and extended Kalman filter (EKF) are adopted to estimate and update the model parameters and states jointly [22]. However, most of these methods are based on current observation batteries, which results in poor generalization performance of the models. Moreover, these methods have limited modeling ability for complex nonlinear processes.

Kernel-based technologies such as support vector machine (SVM) and relevance vector machine (RVM) are also widely used. Weng et al. proposed a method based on experimental battery aging data [23]. They developed a model parameterization and adaptation framework utilizing the simple structure of support vector regression (SVR) representation with determined support vectors (SVs) so that the model parameters can be estimated in real time. Nuhic et al. used SVM to embed diagnosis and prognostics of system health with an aim to estimate the SOH and

RUL of lithium-ion batteries [24]. However, there are some limitations in SVM itself, such as that the kernel function must satisfy Mercer condition, it is easy to fall into local optimum, and it is difficult to determine the loss function and penalty factor. Qin et al. utilized RVM to quantify the relationship between monitoring parameters and capacity data [25]. Moreover, feature vector selection (FVS) is used to remove redundant points in the input data. Thus, the new method has better RUL prediction accuracy and higher sparsity compared to RVM. Compared with SVM, RVM has higher accuracy and lower computational complexity. However, due to the high sparsity, the prediction results of RVM-based methods are less stable.

Nowadays, more and more researchers try to use neural network (NN) to build model. Ren et al. proposed a lithium-ion battery RUL prediction method which combined autoencoder with deep neural network [26]. Because of the superiority in time series prediction, recurrent neural networks (RNN) and long short-term memory (LSTM) have been applied in RUL prediction of the lithium-ion battery. Liu et al. proposed an adaptive recurrent neural network (RNN) for system dynamic state forecasting for the purpose of predicting the RUL of lithium-ion batteries [27]. Due to the problem of gradient vanish and gradient exploding, traditional RNN is difficult to deal with long-term dependence in practice. The appearance of long short-term memory (LSTM) solves this problem well. Zhang et al. utilized LSTM to build RUL model of the lithium-ion battery [28]. The above methods have achieved good results, but there are still some problems that need to be improved. Firstly, methods based on RNN and LSTM usually use time sliding window as input. However, the features in a sliding window have a different effect on the results. We need to find an appropriate way to identify this difference. Secondly, it has been proved that the key parameters have a great influence on the performance of the neural network. we need to find a method to get the best key parameters. Thirdly, in order to improve the performance of the model, we should update the model dynamically with the latest data. In this paper, we proposed a method which can solve these problems better.

III. PRELIMINARIES

A. DATASET

The method proposed in this paper is validated by using the lithium-ion battery RUL dataset from the data repository of the NASA Ames Prognostics Center of Excellence (PCoE) [20]. This paper chooses three battery life cycle datasets: B0005, B0006 and B0018. Each dataset records information about charging, discharging and impedance of batteries. The charging and discharging processes of all batteries are carried out at room temperature. The charging and discharging processes are as follows. Charging was carried out in constant current (CC) mode at 1.5A until the battery voltage reached 4.2V and then continued in a constant voltage (CV) mode until the charge current dropped to 20mA. The discharge was carried out at a constant current (CC) level of 2A until the battery voltage fell to 2.7V, 2.5V and 2.5V for batteries B0005, B0006 and B0018 respectively.

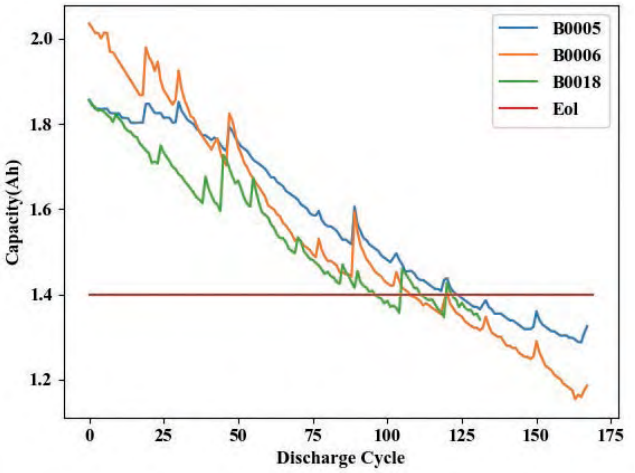


FIGURE 1. Capacity and failure threshold of lithium-ion batteries.

B. PROBLEM STATEMENT

The description file of the dataset indicates that when the capacitance of the battery drops to 70% of the nominal capacitance (2 Ah), the end point of the battery's life is determined. Since the capacity of batteries can only be measured in the discharge cycle data, the declining trend of capacitance and the failure threshold of batteries are shown in Figure 1.

The main objectives of the proposed method are as follows:

1) Realizing online SOH monitoring of the lithium-ion battery.
2) Predicting the RUL of the lithium-ion battery based on historical data.

Among them, SOH is a variable that decreases with time. Because the capacitance of lithium-ion batteries can directly reflect the deterioration of battery performance. Therefore, in this paper, SOH is defined as follows:

$$SOH = \frac{C_t}{C_m} \tag{1}$$

In (1), C_t denote the capacitance of *t-th* cycle, C_m represents the nominal capacitance of the battery. At the same time, the failure threshold of the battery is set to 70% of the nominal capacity of the battery. The definition of EOL is as follows:

$$EOL = C_m * 0.7 = 1.4Ah \tag{2}$$

When online health monitoring is carried out, the next SOH value is predicted mainly by the previous SOH observation values. The problem of online health monitoring is as in (3):

$$SOH^p_{t+1} = f([SOH^r_t, SOH^r_{t-1}, ...SOH^r_{t-w+1}]) \tag{3}$$

In (3), SOH^p_t is the prediction value of step t, SOH^r_t is the observation value of step t, w is the length of slide window. The purpose of our method is to find such a function f. In this way, the SOH of the next moment can be predicted at any time of the battery life cycle.

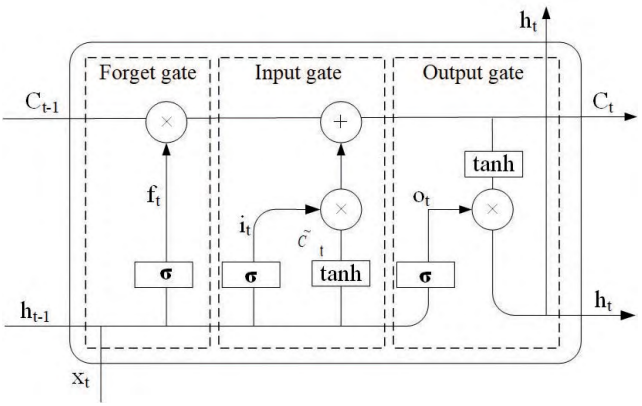


FIGURE 2. Architecture of the LSTM cell.

When the RUL prediction is carried out, the prediction model is established through the whole life cycle history data of other batteries, and then the model is applied to multi-step prediction. When predicting the SOH of step t, assuming that the length of sliding window is w, the number of real values in the previous sliding window is k, and the SOH value of step t are as in (4) and (5):

$$M(\cdot) = f(B_1, B_2..., B_n) \tag{4}$$

$$SOH_t^p = M([SOH_{t-1}^p..., SOH_{t-w+k}^p, SOH_{t-w+k-1}^r, ...SOH_{t-w}^r]) \tag{5}$$

In (4), B_n is the historical life cycle data of other lithium-ion batteries, and M is the model trained by the historical data. Based on this, the SOH value at *n-th* charging and discharging cycles can be predicted. If the predicted SOH reaches the EOL value, the RUL of the battery is as in (6):

$$RUL = n - t \tag{6}$$

In (6), t is the current charge-discharge cycle of the battery.

C. BASIC ALGORITHM THEORY

1) LSTM

Long short-term memory (LSTM) is a special kind of RNN which performs well in the field of time series prediction. Unlike original RNN, LSTM has a gate structure that allows information to pass through selectively [29]. The structure of LSTM is shown in Figure 2.

Through a sigmoid layer and dot products, the gate structure makes it possible to selectively transfer information. There are three kinds of gates, i.e., forget gate, input gate, and the output gate [28]. Based on this, LSTM can store and transfer cell states. This design makes the network composed of LSTM cells have the ability of long-term memory. Therefore, LSTM neural network is used to extract features from the sliding window in this paper.

2) ATTENTION MECHANISM

In recent years, inspired by the attention mechanism of human brain, attention mechanism has been widely used in neural networks and achieved good results [30]–[32]. In many neural networks, the output is usually determined by a sequence of features. For example, the RUL prediction of the lithium-ion battery based on LSTM is usually achieved by the sliding window. The prediction results depend on the features extracted from the sliding window. However, the effect of each feature on the result is different. The main function of attention mechanism is to learn an attention weight of each feature from the sequence and then merge the features according to the attention weight. By introducing a self-attention mechanism [33], the distraction problem can be effectively reduced, and the accuracy of the model will improve.

3) PARTICLE SWARM OPTIMIZATION

Particle swarm optimization (PSO) is a population-based stochastic optimization technique, which was developed by Kennedy and Eberhart [34]. Particle swarm optimization (PSO) searches for global optimum by simulating the predatory behavior of fish/birds. In PSO, each single candidate solution can be regarded as a particle in solution space. The flying process of a particle is the searching process of the particle. Particles have only two properties: speed and location. The optimal solution that each particle searches independently is called individual extremum and the optimal individual extremum in the particle swarm is the current global optimal solution. Finally, by constantly updating speed and location, the optimal solution satisfying the termination condition is obtained. Compared with other optimization algorithms, PSO has fewer parameters to adjust and faster convergence speed [35]. In this paper, we use PSO to optimize parameters.

4) CEEMDAN

Due to the complex physical and chemical changes in the battery and the influence of environmental factors, a large amount of noise inevitably exists in the SOH measurements. In order to effectively eliminate the noise, Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN) is adopted [36], CEEMDAN is a non-stationary signal analysis method. Compared with the traditional empirical modal analysis method, CEEMDAN adds positive and negative white noise with the same amplitude and opposite phase to the original signal. Then, empirical mode decomposition (EMD) is used to get the final decomposition result by calculating the average value. While avoiding modal mixing, CEEMDAN can effectively solve the incomplete decomposition problem of EEMD method [36]. It has the advantages of strong anti-noise ability and fast calculation speed. It has been widely used in time series prediction with noise [37]–[39].

IV. METHODOLOGY

In this section, we first propose a method to realize the SOH monitoring and RUL prediction of the lithium-ion battery. Then, we introduce how to use the proposed PA-LSTM method to establish the basic model. Finally, we describe how to use CEEMDAN to denoise raw data when predicting

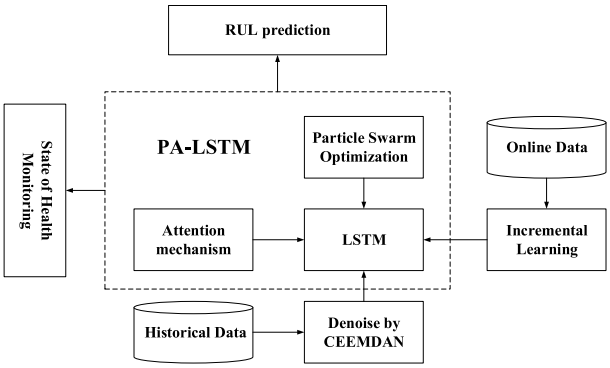


FIGURE 3. The framework for PHM of the lithium-ion battery.

the RUL of the lithium-ion battery, and how to combine incremental learning method to achieve SOH monitoring.

A. A FRAMEWORK FOR PHM OF THE LITHIUM-ION BATTERY

In this paper, we propose a framework which can realize SOH monitoring and RUL prediction of the lithium-ion battery. This framework is shown in Figure 3.

According to the different application requirements, the lithium-ion battery data can be divided into two kinds. The first kind is online data, which is the current measured SOH data of lithium-ion batteries obtained directly from sensors. The other kind is historical data, which is the life cycle battery capacitance data of the same type of lithium-ion batteries under the same working conditions. When monitoring the SOH of lithium-ion batteries, the proposed method first establishes a basic model using the PA-LSTM method and historical data. Then, while using the basic model to achieve SOH monitoring, the model will be updated dynamically by using online data. The output is the SOH value of the next moment. An alarm will be issued if the value is below the failure threshold. When predicting the RUL of the lithium-ion battery, firstly, the raw data will be denoised by CEEMDAN. Then, the denoised data and PA-LSTM method are used to build model. The output of the model is the SOH value of the next moment. Then, the SOH value will be re-entered into the model. The RUL of the battery can be obtained by cycling prediction until the SOH value is lower than EOL.

B. BUILD MODEL WITH PA-LSTM

By introducing PSO and attention mechanism to LSTM neural network, we proposed a method called PA-LSTM. PA-LSTM are used to build model for SOH monitoring and RUL prediction of the lithium-ion battery. The process of using PA-LSTM to train the model is shown in Figure 4.

As shown in Figure 4, LSTM layer is used to extract features from SOH data. After that, we use an attention layer to calculate the attention weights of features. Meanwhile, It has been proved that the parameters of the neural network have a great influence on the training results of the model [40]. Therefore, we use PSO to optimize the key parameters and pre-training the model of the neural network. Based on the above analysis, the key steps of the training process are as follows.

1) FEATURE EXTRACTION

The SOH data are organized by sliding window. Assuming that x_t is the $t-th$ value of a sliding window. In PA-LSTM, we use an LSTM layer to extract features, which is shown as follows:

a: DISCARD USELESS INFORMATION

The first step is to use the forget gate to determine how much information of last cell state C_{t-1} will be stored in the current cell state C_t. There are three types of input data of forget gate: the cell state C_{t-1} of the last step, the hidden state h_{t-1} of the last step, and the current input x_t. Forget gate outputs a sequence of 0 and 1 by receiving h_{t-1} and x_t. This sequence is used for computing with C_{t-1}. 0 represents discarded information and 1 represents retained information. σ is a sigmoid function. The forget gate is calculated as follows:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{7}$$

b: CALCULATING NEW CELL STATE

Input gate is used to determine how much new information can be add to LSTM cell state. There are two parts of input gate: a sigmoid layer, i.e., i_t, which can determine what information should be update, and a tanh layer which can generate a vector C_t^T for updating. The equations to calculate the two outputs are as in (8) and (9):

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{8}$$
$$C_t^T = tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{9}$$

Then, C_{t-1} is multiplied by the f_t which is the result of forget gate and then the product of i_t and C_t is added. A new state value C_t can be obtained as in (10):

$$C_t = f_t * C_{t-1} + i_t * C_t^T \tag{10}$$

c: OUTPUT RESULT

The final output of LSTM Cell is determined by the output gate. There are two kinds of output value. One is current state C_t and the other is current hidden state h_t. The computational process is as follows: First, we run a sigmoid function to get o_t. o_t is used to determine which parts will be output. Then a tanh method will be used to handle current cell state C_t. Finally, multiply it with o_t so that we can get the final result. The equations are as in (11) and (12):

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \tag{11}$$
$$h_t = o_t * tanh(C_t) \tag{12}$$

Through the above steps, we can get the extracted features of the $t-th$ input value x_t.

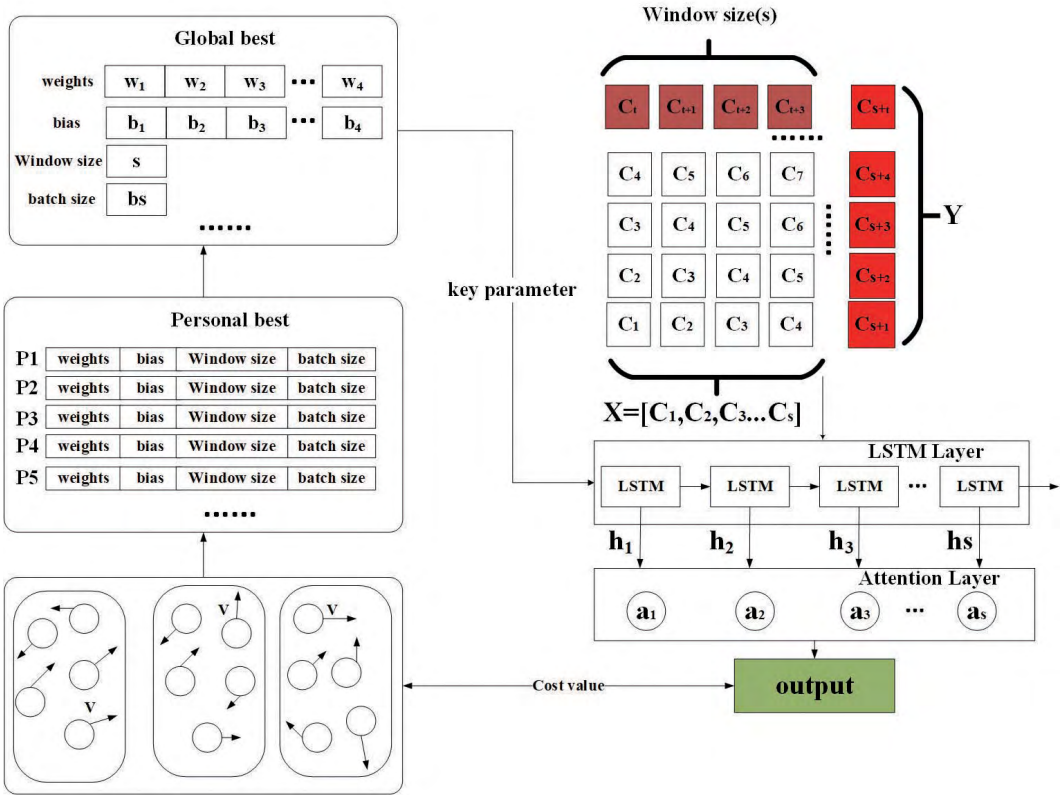


FIGURE 4. Train a model with PA-LSTM.

2) ATTENTION MECHANISM

When analyzing the downtrend of SOH, the effect of each feature on the final result is different. In order to better identify this difference and improve the accuracy of the model, we introduced attention mechanism. According to the description in [33], assuming that the current cycle is t, it is necessary to obtain the output matrix of LSTM layer h_t. Then, we can calculate the corresponding attention weight a_i of h_t. Computational process are as in (13) and (14):

$$e_{ti} = f_{att}(h_t) \tag{13}$$

$$a_i = \frac{exp(e_{ti})}{\sum_{k=1}^{L} exp(e_{tk}) + \varepsilon} \tag{14}$$

In (13) and (14), f_{att} is tanh in this paper. L is the size of h_t. ε is a very small value, which can avoid dividing by zero.

Attention mechanism sets a weight to each feature, which makes features that have a greater impact on the results have a higher weight. The Attention mechanism is shown in the right part of Figure 4. As shown in Figure 4, when the LSTM layer got an input $X = [C_1, C_2, C_3, ..., C_s]$, it will output a sequence $H = [h_1, h_2, h_3, ..., h_s]$. Then, the H will be input to an attention layer. As mentioned in (13) and (14), the attention layer will calculate a a_i for each h_i. Finally, the calculation of output value o is as in (15):

$$o = \sum_{i}^{s} a_i h_i \tag{15}$$

3) PARAMETERS OPTIMIZATION

In this paper, the main functions of particle swarm optimization (PSO) are as follows:

1) The key parameters of the neural network, such as sliding window size, number of hidden units and batchsize, are automatically optimized to avoid the error caused by manual adjustment.
2) Pre-training the neural network while optimizing the key parameters.

As shown in Figure 3, before the model is formally trained, the PSO algorithm is used to optimize the key parameters and pre-train the model. The optimization objectives include batchsize, number of hidden units, window size, weights and biases of each layer, etc. Each point in the solution space is initialized as a D-dimensional particle in PSO algorithm. Then the optimal solution is found by iteration. The particle updates itself by tracking two extremes in each iteration. One is the optimal solution *pBest* of the particle itself. The other is the optimal solution of the whole population, i.e., the global extremum *gBest*. The process of using PSO to optimize parameters is as follows:

Step 1: Calculate the solution space dimension D according to the network structure and the number of parameters to be optimized. Assuming that the weights of each layer of the neural network are W_n, the number of biases is b_n, and the parameters to be optimized are batchsize, windows size and

the number of hidden units. The equation to calculate D is as in (16):

$$D = W_n + b_n + 1 + 1 + 1 \tag{16}$$

Step 2: Initialize PSO. Assuming that N particles form a swarm in D-dimensional solution space. Among them, each particle is a D-dimensional vector is as in (17):

$$X_i = (x_{i1}, x_{i2}, x_{i3}, ..., x_{iD}), \quad i = 1, 2, ..., N \tag{17}$$

According to the pre-set restrictions, each particle will be randomly assigned. Then, a flight velocity is randomly initialized for each particle. The flight velocity of each particle is as in (18):

$$V_i = (v_{i1}, v_{i2}, v_{i3}, ..., v_{iD}), \quad i = 1, 2, ..., N \tag{18}$$

Step 3: Establish optimization objectives. In order to obtain the optimal parameters of the network. The objective of particle swarm optimization is set to be the same as model training, i.e., minimize the loss value.

Step 4: Update particle position and velocity. In order to obtain *pBest* and *gBest*, the equations for updating velocity and location of particles are as in (19) and (20):

$$V_{t+1} = w \cdot v_t + c_1 rand()(pbest_t - p_t) + c_2 rand()(gbest_t - p_t) \tag{19}$$

$$p_{t+1} = p_t + v_{t+1} \tag{20}$$

In (19), w is the inertia value of velocity, c_1 and c_2 represent the extent to which the particle is affected by the individual and global optimal solution, respectively. *rand*() is a function that generates random values between 0 and 1. During the whole process, the optimal solution of each particle and the global optimal solution are updated in real time until the stopping condition is reached (the loss value of the model is no longer decrease, or reach the maximum number of iteration).

C. DONISING BY CEEMDAN

In order to increase the accuracy of RUL prediction, we use CEEMDAN to eliminate noise. The specific processing procedure is as follows: The signal can be decomposed into k intrinsic mode functions (IMF), each IMF is represented by IMF_k. The calculation of the *j-th* IMF of a given signal generated by EMD method is expressed in $E_j()$. Assuming that $s(n)$ is the original signal, in this paper, it is historical SOH data of the lithium-ion battery. CEEMDAN adds a Gaussian white noise $\omega(n)$ which satisfies the standard normal distribution to $s(n)$. After that, perform the following steps:

Step 1: Calculate IMF_1. The computation process of IMF_1 is the same as EEMD [41]. The signal $s(n)+\varepsilon_0\omega_i(n)$ is decomposed I times, where the parameter ε controls the signal-to-noise ratio between the additional noise and the original signal. The calculation method of IMF_1 is as in (21):

$$IMF_1(n) = \frac{1}{I}\sum_{i=1}^{I} IMF_{i1}(n) \tag{21}$$

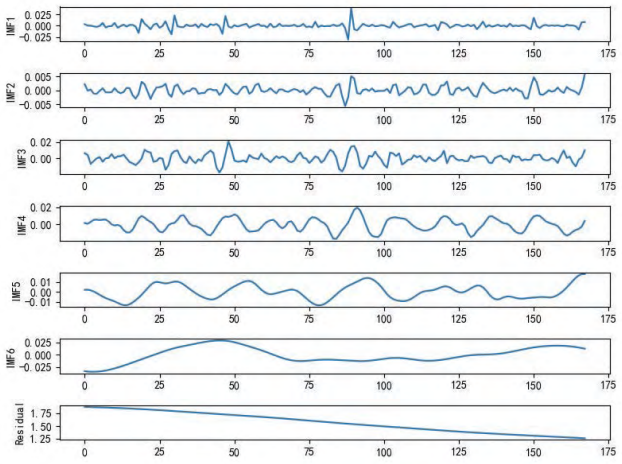


FIGURE 5. Generating IMFs and residual by CEEMDAN.

Step 2: Calculate residual. Calculating method of residual $r_1(n)$ when k equals 1 is as in (22):

$$r_1(n) = s(n) - IMF_1(n) \tag{22}$$

Step 3: Before the next decomposition, obtain the value of the first component from EMD decomposition of white noise, then add it to residual signal to eliminate the error caused by noise to the original signal. The signal to be decomposed is updated to $r_1(n) + \varepsilon_1 E_1(\omega_i(n))(i = 1, 2, ..., I)$. Repeat the calculation of step 2 to get IMF_2, then the computational method of IMF_2 is as in (23):

$$IMF_2(n) = \frac{1}{I}\sum_{i=1}^{I} E_1(r_1 + \varepsilon_1 E_1(\omega^i(n))) \tag{23}$$

Step 4: When k = 2, 3, ..., K, The *k-th* residual is $r_k(n) = r_{k-1}(n) - IMF_k(n)$.

Step 5: Decompose $r_k(n) + \varepsilon_k E_k(\omega_i(n))(i = 1, 2, ..., I)$ until the first EMD component is obtained. The IMF_{k+1} is as in (24):

$$IMF_{k+1}(n) = \frac{1}{I}\sum_{I}^{i=1} E_1(r_k(n) + \varepsilon_k E_k(\omega^i(n))) \tag{24}$$

Step 6: Repeat steps 4-6 until the residual signal cannot be decomposed again, that is, there is at most one extremum for the residual signal.

Finally, the original signal $s(n)$ can be expressed as a combination of k IMFs and a residual $r(n)$, as in (25):

$$s(n) = \sum_{k=1}^{K} IMF_k + r(n) \tag{25}$$

Based on the above description, we select the SOH data of battery B0005 as an example. The results of CEEMDAN decomposition are shown in Figure 5.

We calculated the Pearson correlation coefficients between the decomposed components and the raw data. The calculation results of three lithium-ion batteries, i.e., B0005, B0006 and B0018, are shown in Table 1.

TABLE 1. Pearson correlation between decomposed results and raw data.

IMF	B0005	B0006	B0018
IMF1	0.0248	0.0428	0.0773
IMF2	-0.0078	0.0197	0.0687
IMF3	0.0315	0.0336	0.114
IMF4	0.1183	0.1553	0.0738
IMF5	0.012	-0.0787	0.3028
IMF6	-0.133	0.0149	-0.0159
Residual	0.9932	0.9915	0.9799

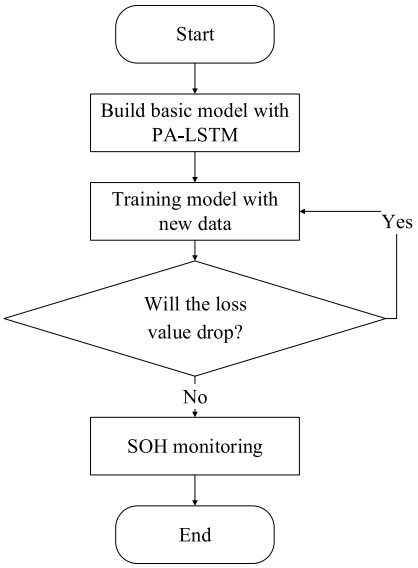


FIGURE 6. Flowchart of SOH monitoring.

From Table 1 we see that the Pearson correlation coefficient of residual is much larger than other components, indicating that residual can reflect the downtrend of SOH. Therefore, in this paper, we regard residual as denoising data for RUL prediction.

D. USE INCREMENTAL LEARNING TO IMPLEMENT SOH MONITORING

When monitoring the SOH, only the SOH value of the next charge-discharge cycle needs to be predicted. Therefore, the model should be able to update dynamically according to the new SOH value. In order to make full use of the latest data to improve the accuracy of the model, an incremental learning mechanism is introduced in the realization of SOH monitoring. Specifically, when the SOH monitoring is started, we use PA-LSTM to build a basic model, and then the basic model is updated whenever a real SOH value is obtained. The condition for terminating the update is that the loss value no longer continues to decrease. Finally, the new model is used to predict the next SOH value. If the predicted value is lower than EOL, an alarm will be issued, otherwise, the monitoring will continue. The overall process of SOH monitoring with incremental learning mechanism is shown in Figure 6.

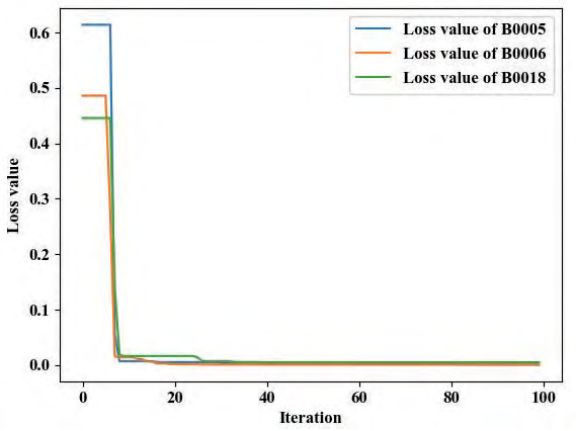


FIGURE 7. Trend of loss values on B0005, B0006 and B0018.

V. RESULTS AND DISCUSSION

In order to validate the proposed method, we use a real dataset of lithium-ion batteries cycle life from NASA [20]. The proposed method is applied to B0005, B0006 and B0018. To evaluate the performance of our method. The SOH monitoring and RUL prediction of these three lithium-ion batteries are carried out respectively and compared with the baseline methods RNN, LSTM and RVM. Python 3.5 is used to program the method. All the experiments were carried out on a laptop, which was configured as follows: CPU is i5-8400, memory is 8G, and the graphics card is NIVIDA 1050Ti 4G.

A. TRAINING MODEL WITH PA-LSTM

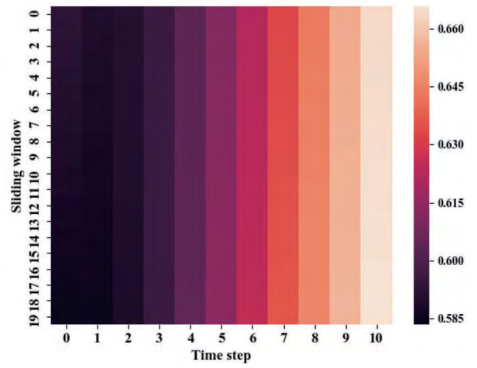
We use PA-LSTM to train the model. Our model consists of two layers of neural networks. The first layer is an LSTM layer composed of 64 neurons and the second layer is an attention layer. The optimizer is stochastic gradient descent (SGD), and the learning rate is 0.02. During the training, some historical data are selected to build the model.

1) PARAMETER OPTIMIZATION AND PRE-TRAINING

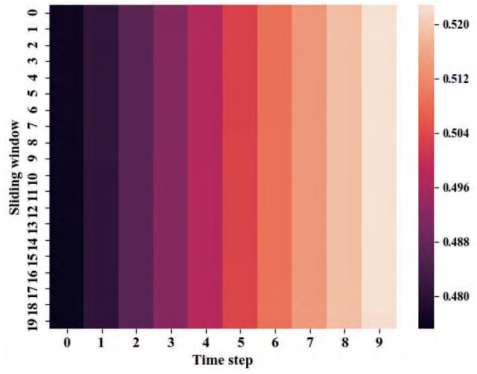
We use the PSO algorithm to optimize the key parameters and pre-train the model. As we introduced in Section III.B, the parameters of PSO include c_1, c_2 and w. These parameters are 0.5, 0.3 and 0.9, respectively. The key parameters of the model that need to be optimized include the number of hidden units (H), window size (W), batchsize (B) and weights as well as biases of every layer of the network. Assuming that the length of train set is $L \in N$, the search range of parameters are $H \in \{8, 16, 32, 64, 128, 256\}$, $W \in [1, 30]$ and $B \in [1, L]$. In order to evaluate the performance of the model, RMSE is used as the evaluation criterion of the model. The calculation of RMSE is as in (26).

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\tilde{y}_t^i - y_t^i)^2} \tag{26}$$

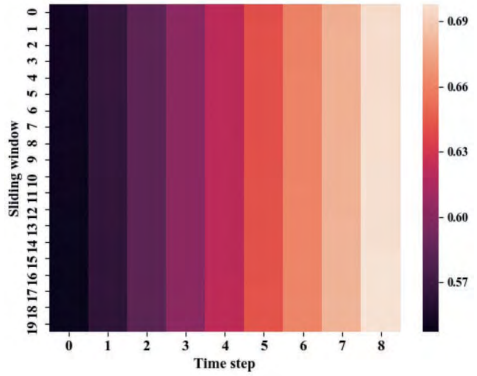
where $\tilde{y}_t$ is the predicted value of the *t-th* charge and discharge cycle, y_t is the measured value of *t-th* charge and discharge



(a) Attention distribution of B0005.



(b) Attention distribution of B0006.



(c) Attention distribution of B0018.

FIGURE 8. Plot of the attention distribution.

cycle, N is the total number of predicted values. When optimizing, the number of particles is 200 and the number of iterations is 100. The trend of loss value obtained in each iteration is shown in Figure 7.

Figure 7 shows that with the increasing of iterations, the loss values of the model shows a downward trend, which proves that PSO can find the appropriate model parameters. The optimized parameters of the model on B0005, B0006 and B0018 is shown in Table 2:

2) TRAIN THE ATTENTION LAYER

In order to verify the effectiveness of attention mechanism on the model, the attention distribution for each dataset is expressed in the form of a thermodynamic chart, which is shown in Figure 8.

TABLE 2. Parameters of the model on every dataset.

Dataset	Hidden units	Batchsize	Window size
B0005	64	20	11
B0006	64	35	10
B0018	64	29	9

TABLE 3. The effect of attention mechanism on result.

Dataset	With attention mechanism	Without attention mechanism
B0005	0.0163	0.0197
B0006	0.0198	0.0286
B0018	0.0294	0.0383

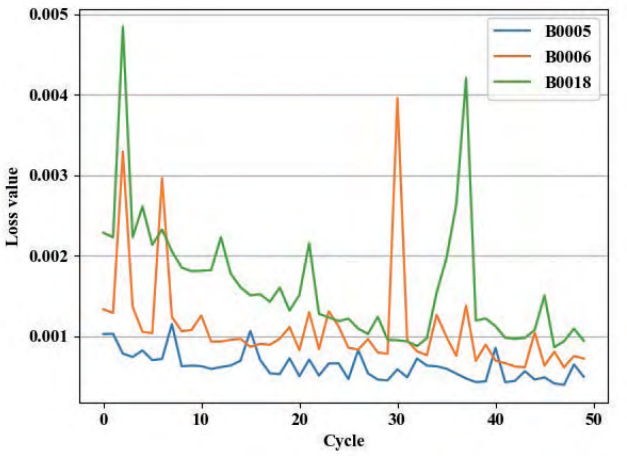


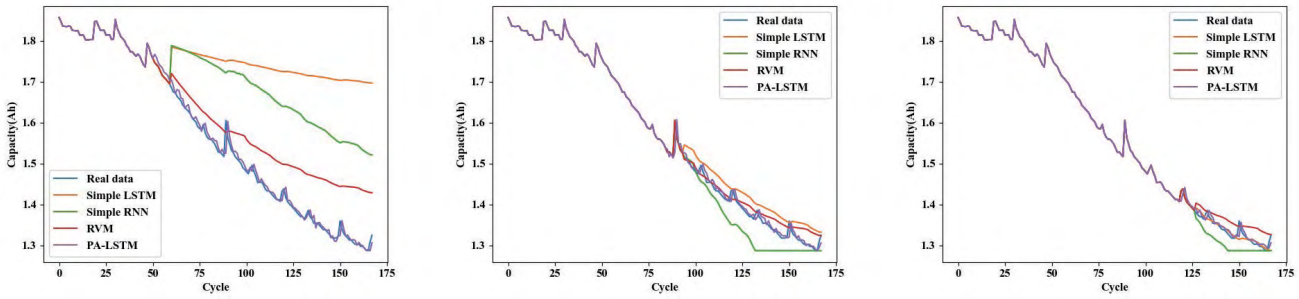
FIGURE 9. Incremental learning process of B0005, B0006 and B0018.

From Figure 8, it can be seen that the attention weight of each feature in a sliding window is different. To further illustrate the effect of attention mechanism on model accuracy, we compared the RMSE values of the model with and without attention mechanism. For each dataset, 50% of the data is the train set and the rest of the data is test set. The effect of attention mechanism on the result is shown in Table 3.

From Table 3, we can find that models with attention mechanism have higher accuracy than those model without attention mechanism. In summary, by adding an attention mechanism, the model assigns weights to the features in a sliding window, which can improve the performance of the model.

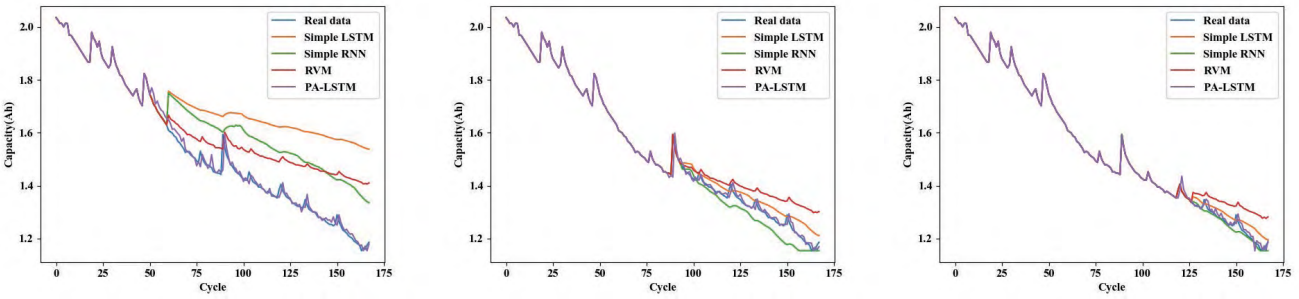
B. ONLINE SOH MONITORING

This section discusses the performance of PA-LSTM method in SOH online monitoring. In this paper, an incremental learning mechanism is introduced in SOH online monitoring. That is, when we obtained a new SOH value, the original model will be updated. This ensures that this method can effectively use the latest data to improve the accuracy of the model. To verify the superiority of incremental learning, we select the last 50 cycles of each dataset for incremental learning,



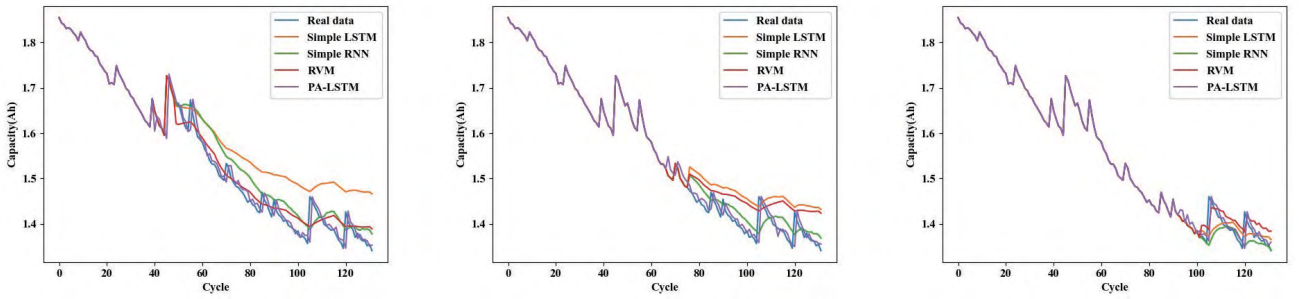
(a) Start at 30% cycle of B0005. (b) Start at 50% cycle of B0005. (c) Start at 70% cycle of B0005.

FIGURE 10. SOH monitoring of B0005.



(a) Start at 30% cycle of B0006. (b) Start at 50% cycle of B0006. (c) Start at 70% cycle of B0006.

FIGURE 11. SOH monitoring of B0006.



(a) Start at 30% cycle of B0018. (b) Start at 50% cycle of B0018. (c) Start at 70% cycle of B0018.

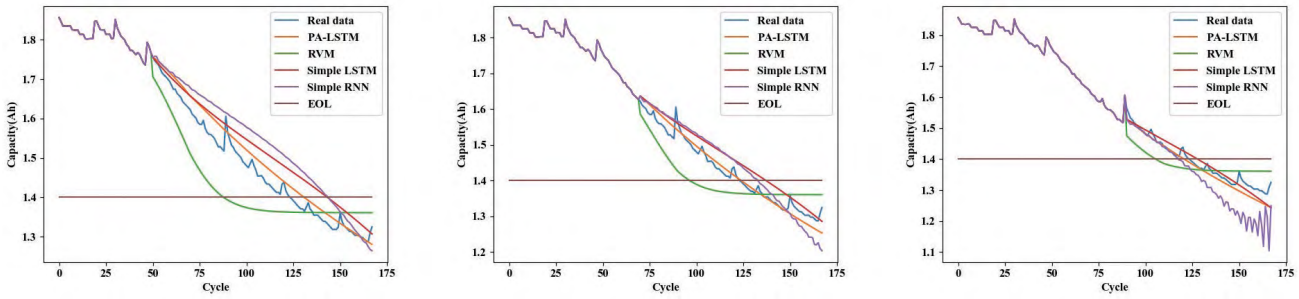
FIGURE 12. SOH monitoring of B0018.

TABLE 4. RMSE of different methods.

	B0005			B0006			B0018		
	30%	50%	70%	30%	50%	70%	30%	50%	70%
Simple LSTM	0.2386	0.0248	0.0061	0.2117	0.0254	0.0161	0.0654	0.0416	0.0142
Simple RNN	0.1708	0.0301	0.0165	0.1349	0.0281	0.0089	0.0331	0.0193	0.0166
RVM	0.0731	0.0139	0.0141	0.1148	0.0509	0.0391	0.0215	0.0347	0.0147
PA-LSTM	0.0119	0.011	0.006	0.0197	0.0159	0.0079	0.0208	0.0152	0.0124

and other data is used to train the base model. Then, every cycle we will update the base model with the latest data. The loss value of the model during incremental learning is shown in Figure 9.

From Figure 9, we can see that loss values show a downward trend. This proves that incremental learning mechanism can improve the accuracy of the model dynamically.



(a) Start at 50-th cycle (b) Start at 70-th cycle. (c) Start at 90-th cycle.

FIGURE 13. RUL prediction of B0005.

In online SOH monitoring, when to start the monitor is important. To verify the effectiveness of PA-LSTM at different start cycles, we compared PA-LSTM with the baseline methods such as RNN, LSTM and RVM. 30%, 50% and 70% of the raw data are selected as train sets, and the monitoring begins from the charging and discharging cycle at the end of the train set. The results of PA-LSTM running on B0005, B0006 and B0018 datasets are shown in Figure 10, Figure 11 and Figure 12. The results demonstrate that: (1) For all of the datasets, the curve predicted by PA-LSTM is closest to the real data curve. (2) For all methods, the more data in the train set, the more accurate the model is. The RMSE values of each method on different datasets and different start cycles are shown in Table 4.

Because both SOH monitoring and RUL prediction use PA-LSTM method, the excellent performance of PA-LSTM method in SOH monitoring means that there will be a very small error when predicting the next value, which lays a good foundation for RUL prediction.

C. RUL PREDICTION

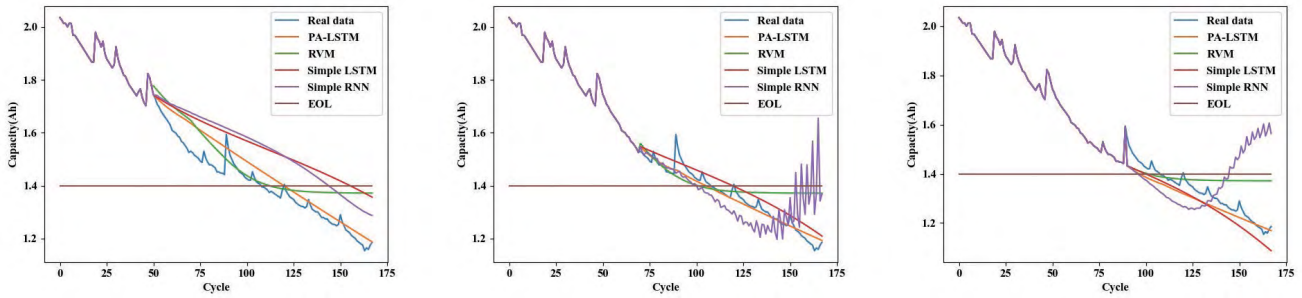
This section mainly verifies the performance of PA-LSTM in predicting the RUL of the lithium-ion battery. Except for the battery to be tested, the other two datasets after CEEMDAN denoising are used as training data. In addition, for the battery to be tested, we set the start cycle, and the data before the start cycle is taken to fine-tune the trained model. In this experiment, we choose RNN, LSTM and RVM for comparison. The criteria for comparison are RMSE, RUL and error of each method.

There are 168 discharge cycles in B0005 data. In the experiment, we select 50, 70 and 90 as the start cycle. The experimental results are shown in Figure 13. The B0006 dataset also has 168 discharge cycles. Similarly, 50, 70 and 90 are selected as the start cycle to predict the RUL of B0006, and Figure 14 shows the results. There are 132 discharge cycles in B0018 data. We select 50, 60 and 70 as the start cycle, respectively. The experimental results are shown in Figure 15.

From Figures 13, 14 and 15, we can see that LSTM-based methods have a better performance than other methods, which proved that LSTM has advantages in dealing with long-term memory. Among them, the prediction curve of PA-LSTM is closest to the real data curve. The EOLs of B0005, B0006 and B0018 predicted by PA-LSTM method

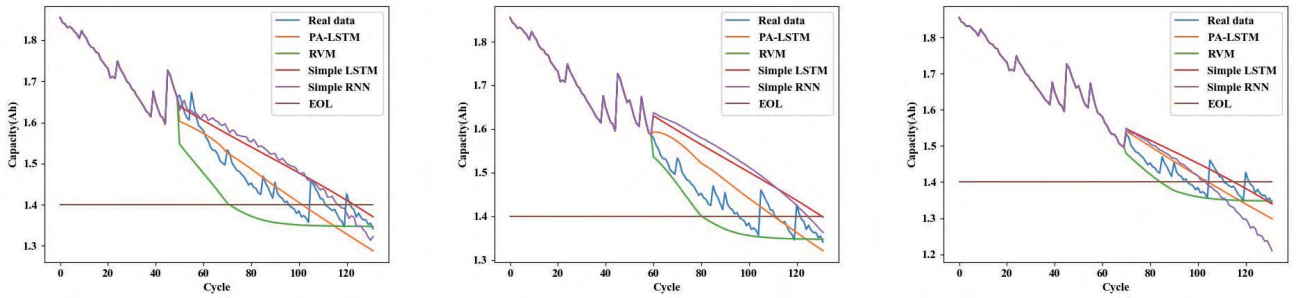
TABLE 5. Comparison of RUL prediction with different methods.

Battery	Start point	Actual RUL	Method	Predicted RUL	Error	RMSE
B0005	50	74	Simple LSTM	94	-20	0.113
			Simple RNN	94	-20	0.1047
			RVM	38	36	0.0784
			PA-LSTM	81	-7	0.0937
	70	54	Simple LSTM	68	-14	0.0299
			Simple RNN	63	-9	0.0331
			RVM	27	27	0.0494
			PA-LSTM	54	0	0.0163
	90	34	Simple LSTM	38	-4	0.0151
			Simple RNN	28	6	0.0484
			RVM	16	18	0.0335
			PA-LSTM	31	3	0.0166
B0006	50	58	Simple LSTM	106	-48	0.1216
			Simple RNN	94	-36	0.1131
			RVM	62	-4	0.0784
			PA-LSTM	70	-12	0.0499
	70	38	Simple LSTM	50	-12	0.0311
			Simple RNN	31	7	0.0799
			RVM	31	7	0.0682
			PA-LSTM	35	3	0.0206
	90	18	Simple LSTM	10	8	0.0352
			Simple RNN	6	12	0.1269
			RVM	11	7	0.0667
			PA-LSTM	7	11	0.0293
B0018	50	46	Simple LSTM	73	-27	0.0498
			Simple RNN	67	-21	0.0551
			RVM	21	25	0.0601
			PA-LSTM	52	-6	0.0324
	60	36	Simple LSTM	71	-35	0.0659
			Simple RNN	65	-29	0.0721
			RVM	21	15	0.034
			PA-LSTM	51	-15	0.0381
	70	26	Simple LSTM	43	-17	0.0327
			Simple RNN	31	-5	0.0466
			RVM	25	1	0.03
			PA-LSTM	34	-8	0.0287



(a) Start at 50-th cycle (b) Start at 70-th cycle. (c) Start at 90-th cycle.

FIGURE 14. RUL prediction of B0006.



(a) Start at 50-th cycle (b) Start at 60-th cycle. (c) Start at 70-th cycle.

FIGURE 15. RUL prediction of B0018.

are 124, 108 and 96, respectively. To further illustrate the advantages of PA-LSTM in predicting RUL of lithium-ion batteries, the detailed values of RUL, error and RMSE of three datasets predicted by each method are shown in Table 5.

The relationship between the predicted RUL (PR), the measured RUL (AR) and the error satisfies the following formula:

$$Error = AR - PR \tag{27}$$

In Table 5, it can be seen that the error and RMSE values of PA-LSTM are relatively small on all datasets. In fact, the average error and RMSE values of LSTM, RNN, RVM and PA-LSTM on different datasets and different start cycles are -19, -11, -15, -3 and 0.0549, 0.0775, 0.0554, 0.0362, respectively. Therefore, PA-LSTM has smaller RMSE and smaller error values than other methods, which proves that the PA-LSTM method proposed in this paper is more stable and accurate in the RUL prediction of the lithium-ion battery.

VI. CONCLUSION

SOH monitoring and RUL prediction are important in PHM of lithium-ion batteries. We proposed a new method named PA-LSTM to solve these two problems by using neural network technology and signal processing methods. The main contributions of this paper are as follows: (1) The key parameters of the neural network have a great influence on the accuracy of the model. In this paper, particle swarm optimization (PSO) is introduced to optimize the weights, biases of the neural network and key parameters of the model. (2) In order to solve the problem of distraction, we introduced the attention mechanism, which sets a weight for each feature according to its impact on the results, so that the method has higher accuracy. (3) In RUL prediction of lithium-ion batteries, CEEMDAN method is used to denoise raw data which aimed to extract the downtrend of SOH. (4) We use incremental learning mechanism to update the model dynamically during SOH monitoring, which can make full use of the latest data to improve the accuracy of the model. The proposed method is validated on NASA lithium-ion batteries datasets. The experimental results show that the proposed method is more accurate than RNN, LSTM and RVM in SOH monitoring and RUL prediction of lithium-ion batteries.

REFERENCES

[1] X. Hu, C. Zou, C. Zhang, and Y. Li, "Technological developments in batteries: A survey of principal roles, types, and management needs," *IEEE Power Energy Mag.*, vol. 15, no. 5, pp. 20–31, Sep. 2017.

[2] X. Gong, R. Xiong, and C. C. Mi, "A data-driven bias-correction-method-based lithium-ion battery modeling approach for electric vehicle applications," *IEEE Trans. Ind. Appl.*, vol. 52, no. 2, pp. 1759–1765, Oct. 2016.

[3] P. J. Swornowski, "Destruction mechanism of the internal structure in lithium-ion batteries used in aviation industry," *Energy*, vol. 122, pp. 779–786, Mar. 2017.

[4] G. Giordano, V. Klass, M. Behm, G. Lindbergh, and J. Sjöberg, "Model-based lithium-ion battery resistance estimation from electric vehicle operating data," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 3720–3728, May 2018.

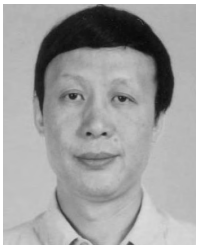
[5] K. Goebel, B. Saha, A. Saxena, J. R. Celaya, and J. P. Christophersen, "Prognostics in battery health management," *IEEE Instrum. Meas. Mag.*, vol. 11, no. 4, pp. 33–40, Aug. 2008.

[6] N. Williard, W. He, C. Hendricks, and M. Pecht, "Lessons learned from the 787 Dreamliner issue on lithium-ion battery reliability," *Energies*, vol. 6, no. 9, pp. 4682–4695, 2013.

[7] A. Lahiri, N. Shah, and C. Dales, "Building a safer, denser lithium-ion battery," *IEEE Spectr.*, vol. 55, no. 3, pp. 34–39, Mar. 2018.
[8] C. Li, Z. Chen, J. Cui, Y. Wang, and F. Zou, "The lithium-ion battery state-of-charge estimation using random forest regression," in *Proc. Prognostics Syst. Health Manage. Conf. (PHM-Hunan)*, Aug. 2014, pp. 336–339.
[9] X. Xu, Z. Li, and N. Chen, "A hierarchical model for lithium-ion battery degradation prediction," *IEEE Trans. Rel.*, vol. 65, no. 1, pp. 310–325, Mar. 2016.
[10] F. Li and J. Xu, "A new prognostics method for state of health estimation of lithium-ion batteries based on a mixture of Gaussian process models and particle filter," *Microelectron. Rel.*, vol. 55, no. 7, pp. 1035–1045, Jun. 2015.
[11] S. Hong, Z. Zhou, E. Zio, and W. Wang, "An adaptive method for health trend prediction of rotating bearings," *Digit. Signal Process.*, vol. 35, pp. 117–123, Dec. 2014.
[12] S. Hong, Z. Zhou, E. Zio, and K. Hong, "Condition assessment for the performance degradation of bearing based on a combinatorial feature extraction method," *Digit. Signal Process.*, vol. 27, pp. 159–166, Apr. 2014.
[13] Y. Liu, H.-Z. Huang, Z. Wang, Y. Li, and Y. Yang, "A joint redundancy and imperfect maintenance strategy optimization for multi-state systems," *IEEE Trans. Rel.*, vol. 62, no. 2, pp. 368–378, Jun. 2013.
[14] Y. Liu, M. J. Zuo, Y.-F. Li, and H. Z. Huang, "Dynamic reliability assessment for multi-state systems utilizing system-level inspection data," *IEEE Trans. Rel.*, vol. 64, no. 4, pp. 1287–1299, Dec. 2015.
[15] T. R. Ashwin, Y. M. Chung, and J. Wang, "Capacity fade modelling of lithium-ion battery under cyclic loading conditions," *J. Power Sources*, vol. 328, pp. 586–598, Oct. 2016.
[16] M. Mishra, J. Martinsson, M. Rantatalo, and G. Kai, "Bayesian hierarchical model-based prognostics for lithium-ion batteries," *Rel. Eng. Syst. Saf.*, vol. 172, pp. 25–35, Apr. 2018.
[17] D. A. Pola, H. F. Navarrete, M. E. Orchard, R. S. Rabié, M. A. Cerda, B. E. Olivares, J. F. Silva, P. A. Espinoza, and A. Pérez, "Particle-filtering-based discharge time prognosis for lithium-ion batteries with a statistical characterization of use profiles," *IEEE Trans. Rel.*, vol. 64, no. 2, pp. 710–720, Jun. 2015.
[18] B. Mo, J. Yu, D. Tang, H. Liu, and J. Yu, "A remaining useful life prediction approach for lithium-ion batteries using Kalman filter and an improved particle filter," in *Proc. IEEE Int. Conf. Prognostics Health Manage. (ICPHM)*, Jun. 2016, pp. 1–5.
[19] H. Zhang, Q. Miao, X. Zhang, and Z. Liu, "An improved unscented particle filter approach for lithium-ion battery remaining useful life prediction," *Microelectron. Rel.*, vol. 81, pp. 288–298, Feb. 2018.
[20] Y. Zhou and M. Huang, "Lithium-ion batteries remaining useful life prediction based on a mixture of empirical mode decomposition and ARIMA model," *Microelectron. Rel.*, vol. 65, pp. 265–273, Oct. 2016.
[21] Y. Zhang, R. Xiong, H. He, and M. Pecht, "Lithium-ion battery remaining useful life prediction with Box–Cox transformation and Monte Carlo simulation," *IEEE Trans. Ind. Electron*, vol. 66, no. 2, pp. 1585–1597, Feb. 2019.
[22] X. Xu and N. Chen, "A state-space-based prognostics model for lithium-ion battery degradation," *Rel. Eng. Syst. Saf.*, vol. 159, pp. 47–57, Mar. 2017.
[23] C. Weng, Y. Cui, J. Sun, and H. Peng, "On-board state of health monitoring of lithium-ion batteries using incremental capacity analysis with support vector regression," *J. Power Sources*, vol. 235, pp. 36–44, Aug. 2013.
[24] A. Nuhic, T. Terzimehic, T. Soczka-Guth, M. Buchholz, and K. Dietmayer, "Health diagnosis and remaining useful life prognostics of lithium-ion batteries using data-driven methods," *J. Power Sources*, vol. 239, pp. 680–688, Oct. 2013.
[25] X. Qin, Q. Zhao, H. Zhao, W. Feng, and X. Guan, "Prognostics of remaining useful life for lithium-ion batteries based on a feature vector selection and relevance vector machine approach," in *Proc. IEEE Int. Conf. Prognostics Health Manage. (ICPHM)*, Jun. 2017, pp. 1–6.
[26] L. Ren, L. Zhao, S. Hong, S. Zhao, H. Wang, and L. Zhang, "Remaining useful life prediction for lithium-ion battery: A deep learning approach," *IEEE Access*, vol. 6, pp. 50587–50598, 2018.
[27] J. Liu, A. Saxena, K. Goebel, B. Saha, and W. Wang, "An adaptive recurrent neural network for remaining useful life prediction of lithium-ion batteries," Dept. Mech. Aerosp. Eng., Carleton Univ., Ottawa, ON, Canada, Tech. Rep. ADA562707, 2010.
[28] Y. Zhang, R. Xiong, H. He, and M. G. Pecht, "Long short-term memory recurrent neural network for remaining useful life prediction of lithium-ion batteries," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 5695–5705, Jul. 2018.
[29] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[30] Y. Zhu, C. Zhao, H. Guo, J. Wang, X. Zhao, and H. Lu, "Attention couplenet: Fully convolutional attention coupling network for object detection," *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 113–126, Jan. 2019.
[31] M. Sun, Z. Zhou, Q. Hu, Z. Wang, and J. Jiang, "SG-FCN: A motion and memory-based deep learning model for video saliency detection," *IEEE Trans. Cybern.*, vol. 49, no. 8, pp. 2900–2911, Aug. 2018.
[32] T. Liu, S. Yu, B. Xu, and H. Yin, "Recurrent networks with attention and convolutional networks for sentence representation and classification," *Appl. Intell.*, vol. 48, pp. 3797–3806, Oct. 2018.
[33] C. Raffel and D. P. W. Ellis, "Feed-forward networks with attention can solve some long-term memory problems," 2015, *arXiv:1512.08756*. [Online]. Available: https://arxiv.org/abs/1512.08756
[34] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. Int. Conf. Neural Netw. (ICNN)*, Dec. 2002, pp. 1942–1948.
[35] J. Kennedy, "Particle swarm optimization," in *Encyclopedia of Machine Learning*, C. Sammut and G. I. Webb, Eds. Boston, MA, USA: Springer, 2011.
[36] M. A. Colominas, G. Schlotthauer, and M. E. Torres, "Improved complete ensemble EMD: A suitable tool for biomedical signal processing," *Biomed. Signal Process. Control*, vol. 14, pp. 19–29, Nov. 2014.
[37] J. Li and Q. Li, "Medium term electricity load forecasting based on CEEMDAN, permutation entropy and ESN with leaky integrator neurons," *Elect. Mach. Control*, vol. 19, pp. 70–80, Aug. 2015.
[38] S. Adarsh and M. J. Reddy, "Multiscale analysis of suspended sediment concentration data from natural channels using the Hilbert-Huang transform," *Aquatic Procedia*, vol. 4, pp. 780–788, Jan. 2015.
[39] A. C. Palaninathan, X. Qiu, and P. N. Suganthan, "Heterogeneous ensemble for power load demand forecasting," in *Proc. IEEE Region 10 Conf. (TENCON)*, Nov. 2016, pp. 2040–2045.
[40] S. Han, J. Pool, J. Tran, and W. J. Dally, "Learning both weights and connections for efficient neural network," in *Proc. 28th Int. Conf. Neural Inf. Process. Syst.*, vol. 1. Cambridge, MA, USA: MIT Press, 2015, pp. 1135–1143.
[41] T. Wang, M. Zhang, and Q. Yu, "Comparing the applications of EMD and EEMD on time–frequency analysis of seismic signal," *J. Appl. Geophys.*, vol. 83, pp. 29–34, Aug. 2012.

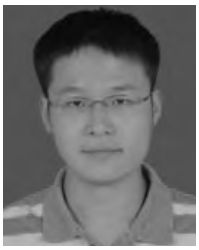
JIANTAO QU received the B.S. and M.S. degrees in computer application technology from Henan University, China, in 2016. He is currently pursuing the Ph.D. degree with the School of Computer and Information Technology, Beijing Jiaotong University.

His research interests include rail transit information technology, prognostic and health management (PHM), and deep learning.



FENG LIU received the B.S. and M.S. degrees from Beijing Jiaotong University, in 1983 and 1988, respectively, and the Ph.D. degree from the Renmin University of China, in 1997.

He is currently a Professor with the School of Computer and Information Technology, Beijing Jiaotong University. He is also the Director of the Engineering Research Center of Network Management Technology for High-Speed Railway, Ministry of Education. He has been involved in research and engineering in the field of rail transit information technology over 20 years. He has published over 60 papers and four books. His research interests include rail transit information technology, software engineering technology, big data for railway, and prognostic and health management (PHM). He is also a Millions of Talent Projects National candidate. He received the First Class of Technological Progress Awards, in 1991.



YUXIANG MA received the B.S. degree in network engineering from Henan University, in 2009, and the Ph.D. degree from the Computer Network Information Center, Chinese Academy of Sciences, in 2019.

He is currently a Lecturer with Henan University. His research interests include the future Internet architecture and technologies, network security, and big data for networking.

JIAMING FAN received the M.S. degree in computer application technology from Henan University, China, in 2015. He is currently pursuing the Ph.D. degree with the School of Computer and Information Technology, Beijing Jiaotong University.

His research interests include rail transit information technology, prognostic and health management (PHM), and big data for railway.