Review

Overview of Machine Learning Methods for Lithium-Ion Battery Remaining Useful Lifetime Prediction

Siyu Jin [1,*], **Xin Sui** [1], **Xinrong Huang** [1], **Shunli Wang** [2], **Remus Teodorescu** [1] and **Daniel-Ioan Stroe** [1]

1   Department of Energy Technology, Aalborg University, DK-9220 Aalborg, Denmark; xin@energy.aau.dk (X.S.); hxi@energy.aau.dk (X.H.); ret@energy.aau.dk (R.T.); dis@energy.aau.dk (D.-I.S.)
2   School of Information Engineering, Southwest University of Science and Technology, Mianyang 621010, China; wangshunli@swust.edu.cn
*   Correspondence: sji@et.aau.dk; Tel.: +45-52677200

Abstract: Lithium-ion batteries play an indispensable role, from portable electronic devices to electric vehicles and home storage systems. Even though they are characterized by superior performance than most other storage technologies, their lifetime is not unlimited and has to be predicted to ensure the economic viability of the battery application. Furthermore, to ensure the optimal battery system operation, the remaining useful lifetime (RUL) prediction has become an essential feature of modern battery management systems (BMSs). Thus, the prediction of RUL of Lithium-ion batteries has become a hot topic for both industry and academia. The purpose of this work is to review, classify, and compare different machine learning (ML)-based methods for the prediction of the RUL of Lithium-ion batteries. First, this article summarizes and classifies various Lithium-ion battery RUL estimation methods that have been proposed in recent years. Secondly, an innovative method was selected for evaluation and compared in terms of accuracy and complexity. DNN is more suitable for RUL prediction due to its strong independent learning ability and generalization ability. In addition, the challenges and prospects of BMS and RUL prediction research are also put forward. Finally, the development of various methods is summarized.

Keywords: Lithium-ion batteries; battery management system; remaining useful lifetime prediction; machine learning

1. Introduction

With the development of the electrification era, the vigorous advancement of new energy vehicles, and the Internet of Things, the importance of energy storage system performance has become prominent. Lithium-ion batteries stand out among various energy storage solutions due to their high energy density, high power capability, and low self-discharge rate [1,2]. At the same time, this also puts forward higher requirements and challenges for the development of battery management technology. A comprehensive battery management system (BMS) should include the following functions: battery data collection, battery status determination and prediction, charge and discharge control, safety protection, thermal management, balance control, and communication [3].

The accuracy of state estimation is an important criterion for evaluating the performance of BMS. A high-performance BMS can make the energy storage system operate reliably and extend the battery lifetime [4]. The optimized BMS should provide multi-task processing capabilities, which can make various tasks work together. At the same time, a real-time operating system is introduced to monitor system parameters and status in real-time so that the system can be adjusted in time.

However, the lifetime of everything is limited, and Lithium-ion batteries are no exception. The price and aging of Lithium-ion batteries are the two main factors that hinder their acceptance in a wider range of applications [5]. The performance of Lithium-ion batteries will decrease with calendar aging and cycle aging, due to various aging

phenomena [6,7]. Battery aging will increase operating costs, reduce the equipment service life, and affect the safe operation of equipment [8–10]. Generally, when the capacity drops to 80% of the initial value, the battery reaches the end of its service life [11]. The remaining useful lifetime (RUL) is defined as the amount of operation time in a certain application until the battery reaches the predefined end-of-life criterion; it represents the period from the present observation to the end of life (EOL) [12–14].

The aging mechanism of the battery is affected by the battery material and the internal chemical reaction during charging and discharging. Battery aging is a complex process of change and presents a highly non-linear trend [15,16]. In actual operation, factors such as operating temperature, charge/discharge rate, and environmental stability will affect the aging process, thereby affecting the battery lifetime [11,17,18].

Therefore, how to accurately predict the RUL under complex working conditions is an important issue to be solved urgently. In the context of industry, RUL predicts that it can reduce investment costs and improve profitability [19,20]. In the academic world, improving the prediction accuracy of RUL can make predictive adjustments to the use of the energy storage system in advance, so that the system can operate safely and stably, and extend the service life of the battery [21,22].

RUL prediction methods can generally be divided into three categories: model-based methods, data-driven methods [23], and hybrid methods [24]. Among them, model-based methods usually include physical models, electrochemical models, and so on. The electrochemical model uses detailed mathematical models to accurately express the internal chemical reaction process of the battery [25–27]. Therefore, these methods will also bring about problems such as high complexity and high calculation cost while achieving high estimation accuracy. In addition, to complete the parameterization process of the electrochemical model, it is usually necessary to disassemble the battery, which will bring great inconvenience in the actual application process [28,29]. A Lithium-ion battery is a highly complex electrochemical power system. When performing RUL prediction for Lithium-ion batteries, model-based methods are usually complicated and difficult to implement, while data-driven methods are more suitable for estimating the RUL of Lithium batteries, especially in the case of a large number of historical data applications [30–32]. Therefore, data-driven prediction methods have attracted much attention.

Among the data-driven methods, ML integrates human knowledge into a machine to achieve previously unavailable functions and performance, and it promotes the interaction between humans and ML systems, making ML decisions understandable to humans [33,34]. When ML technology is applied to predict the RUL of Lithium-ion batteries, it also has the potential of high prediction accuracy and high calculation efficiency [35]. The work in [23] reviewed different methods of Lithium-ion battery RUL prediction, including model-based, data-driven, and hybrid methods. The work in [31] reviewed the application of adaptive filter technology, artificial intelligence technology, stochastic technology, and other related technologies in the prediction of the RUL and state of health (SOH) of electric vehicle batteries. The work in [36] reviewed the use of adaptive methods in the construction of battery aging models, including different characteristics that affect the ability of the model to update itself. This work also classifies the adaptive aging models of Lithium-ion batteries and defines different evaluation criteria in terms of accuracy and computational cost to judge the performance of the model. In [37], the author reviewed the application of ML in energy storage systems. Energy storage devices include batteries, capacitors/supercapacitors, and fuel cells. The work in [38] reviewed the application of intelligent algorithms for battery technology in electric vehicles and evaluated the functions, structure, configuration, accuracy, advantages, and disadvantages of smart algorithms in battery state estimation. The work in [39] paid more attention to the cost-effectiveness and feasibility of the practical application of data-driven methods in the field of Lithium-ion battery health estimation and lifetime prediction. The main ideas of these review papers are shown in Table 1.

Table 1. The main idea of related review papers.

Literature No.	Method		Application Scenario	Contribution
[23]	- - -	Model-based Data-driven Hybrid methods	The prediction of the RUL of Lithium-ion batteries	Reviews recent advances in model-based, data-driven, and hybrid approaches
[31]	- - - -	Adaptive filter technique Intelligent techniques Stochastic technique Other techniques	The RUL and SOH prediction for electric vehicle batteries	Identifies the classifications, characteristics, and evaluation processes with advantages and disadvantages for EV applications
[36]	-	Adaptive method	Battery aging model construction	Classifies adaptive Lithium-ion battery aging modeling methods and defines evaluation methods
[37]	-	ML	Energy storage devices and systems (batteries, capacitors/super-capacitors, fuel cells)	Classifies and briefly describes application examples
[38]	-	Intelligent algorithms	Battery state estimation in electric vehicle	Evaluates the functions, structure, configuration, accuracy, advantages, and disadvantages
[39]	-	Data-driven methods	Lithium-ion battery health assessment and life prediction	The feasibility and cost-effectiveness of improving battery health in practical applications

These reviews mainly focus on how to apply model-based methods to SOH, RUL prediction, and how to apply data-driven methods to the joint prediction of SOH and RUL. Few comments have focused on the application of the ML method in Lithium-ion battery RUL prediction. To make up for this research gap, the author consulted 216 papers, selected 75 papers based on the ML classification for RUL prediction demand, and reviewed the support vector machine, Gaussian process regression, extreme learning machine, deep neural network, and recurrent neural network. A new standard is proposed to evaluate and compare the methods proposed in the literature, starting from the two general directions of accuracy and robustness, to emphasize the detailed information, advantages, and limitations of these methods.

The remainder of this article is organized as follows. Section 2 introduces the basic principles of the support vector machine (SVM), Gaussian process regression (GPR), extreme learning machine (ELM), deep neural network (DNN), and recurrent neural network (RNN) and their application in Lithium-ion battery RUL prediction. Section 3 compares the aforementioned ML methods from the perspective of accuracy and algorithm parameters. Section 4 presents the challenges and prospects for RUL prediction. Section 5 concludes this work.

2. Machine Learning for RUL Prediction

The ML method is the preferred method for predicting RUL when historical life cycle data are available [33,34]. Figure 1 shows the basic workflow of introducing ML in the process of predicting RUL. First, collect raw data that can be directly measured by the battery, such as operating temperature (T), charge/discharge current (I), and operating voltage (V), as inputs for the ML model. Secondly, perform preprocessing operations such as denoising on the original data and extracting the feature vector representing the aging behavior. The feature extraction step seriously affects the RUL estimation performance. Finally, the trained ML model will simulate the relationship between the characteristic value and the battery RUL and realize the prediction of the RUL.
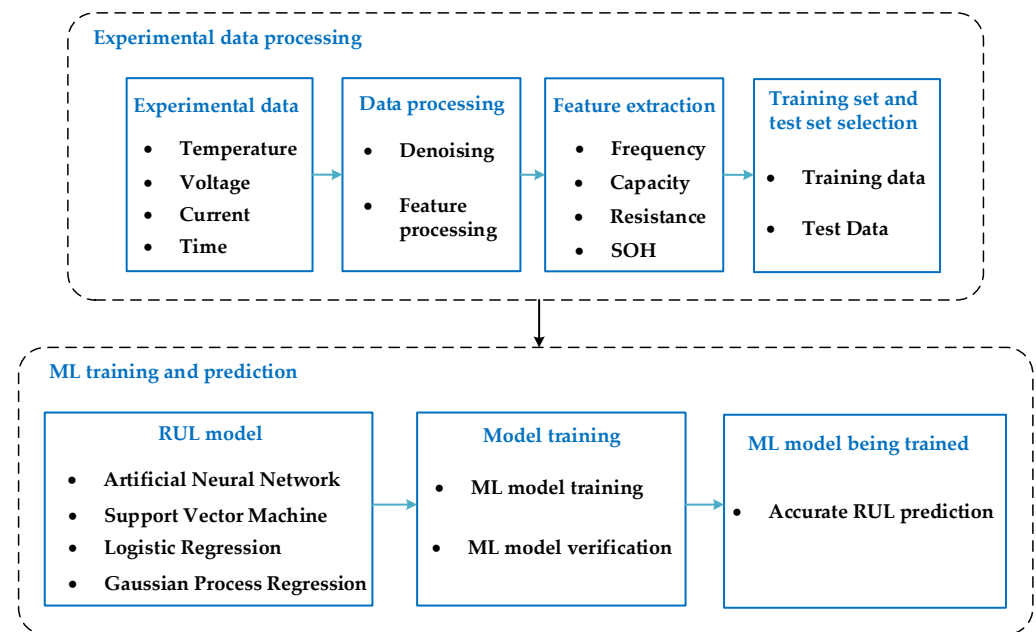
Figure 1. The basic workflow of applying ML to RUL prediction.

2.1. Support Vector Machine

Support vector machines (SVM) have received widespread attention due to their strong advantages in processing small training data sets. SVM is a kernel-based non-parametric ML technology. When the size of the training data set increases, the number of support vectors will increase accordingly. In a complex system, this method can be modeled according to the characteristics of the system and can provide sufficient data support, so it has the characteristics of high flexibility [40].

SVM uses two parallel hyperplanes to clearly classify linearly separable data sets. Equation (1) is the decision boundary between two parallel boundaries, where w is the weight and *b* is the deviation parameter vector. The distance between the decision boundary and each hyperplane is suitable for the standardized data set. SVM introduces the hinge loss function based on the hyperplane to reduce the classification error on the linear inseparable data set.

$$w^T x + b = 0 \tag{1}$$

$$y(x) = \frac{\lambda}{N} \sum_{n=1}^{N} w^T k(x_i, x_j) + b \tag{2}$$

$$w^T \varphi(x) + b = 0 \tag{3}$$

$$k(x_i, x_j) = \varphi(x_i)^T \varphi(x_j) \tag{4}$$

The problem becomes the function $y(x)$ in the minimization Equation (2), where *n* represents the number of samples and λ represents the regularization parameters. Specifically, SVM can also use the kernel function to transform the input low-dimensional vector into a high-dimensional feature space, and then use the hyperplane to separate the data, thereby applying the kernel method (the decision-making boundary element has Equations (3) and (4) as shown above, where φ is the mapping function). The structure of the support vector machine algorithm is shown in Figure 2.
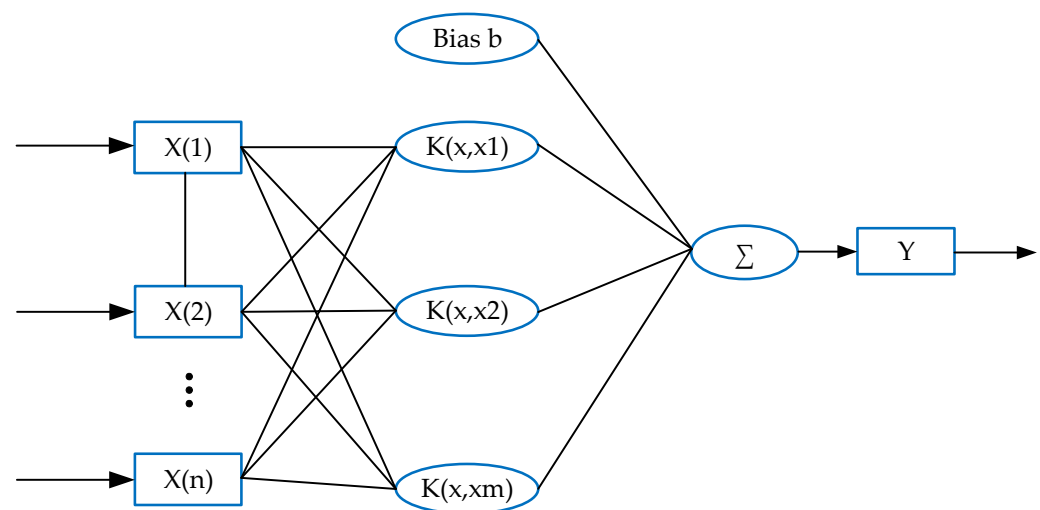
Figure 2. The structure of the support vector machine algorithm.

When SVM is used as a continuous value regression tool, it is called support vector regression (SVR). SVR needs to find a hyperplane, similar to SVM. The difference is that SVM needs to find a hyperplane with the largest gap. In SVR, a threshold ε needs to be defined, and only the loss of the data points in the strip area as shown in Equation (5) is calculated, and then the points outside the area are regressed.

$$|y(x) - y| > \varepsilon \tag{5}$$

Regression is achieved by searching for the smallest marginal fit. SVR is one of the most commonly used regression methods at this stage. In the regression model, the convexity solution of SVR can be obtained by constructing the Lagrangian loss function. At the same time, the mapping function can be used to convert a low-dimensional nonlinear input space into a high-dimensional linear feature space and a nonlinear regression problem into a linear problem. The SVR algorithm proposes a non-parametric regression method, which can be updated through model retraining. Because of its ability to describe the non-linear correlation of input and output data, SVR is suitable for health diagnosis and prediction tasks.

To enhance the stability and robustness of SVR, decrement and increment strategies are used to integrate large-scale training samples to train SVR, and at the same time, uncorrelated data are denoised. However, while enhancing the performance of the model, the calculation time and complexity are increased. Patil et al. [41] used the feature vectors extracted from the voltage and temperature curves as the input data set for RUL prediction and built a prediction model based on SVR. The root means square error (RMSE) of the model is 0.357%. When the confidence interval is 95%, the upper and lower errors are 7.87% and 10.75%, respectively. Zhao et al. [42] calculated the battery capacitance, by varying the different time intervals of the same voltage difference during the charging and discharging process, and combined it with the method of processing the data set when the feature vector is selected to improve the accuracy of SVR. The maximum RMSE of this method is 1%. Du et al. [43] established an SVR-based RUL prediction model for Lithium-ion batteries using six sets of coupled stress experimental data; the relative error of the RUL prediction for 600 cycles is below 5%.

When the SVM method performs RUL prediction, the activation function is usually selected as the radial-based kernel, the training algorithm is the logistic regression function margin, the hyperparameter adjustment methods are the regularization factor, the SVM type regression, and the kernel parameter.

2.2. Gaussian Process Regression

The prediction part is added to the prior knowledge based on the Bayesian framework. Such a kernel-based ML method is Gaussian process regression (GPR). GPR uses the average forecast variance to describe the associated uncertainty. The structure of the GPR algorithm is shown in Figure 3. The GPR model is flexible, non-parametric, and probabilistic. It can be updated through online retraining and has been widely used in prognostic analysis. GPR was used in the initial stage to predict the decay trend of battery internal resistance, and the decay of battery internal resistance is the main factor in the decrease of battery capacity [44]. Therefore, GPR is gradually being applied to the prediction of battery RUL. The attenuation of capacity is a very complicated non-linear process. It is affected by many uncertain environmental factors and working conditions. The improper operation will also lead to the rapid attenuation of capacity. Thus, a single covariance function will lead to unreliable predictions for nonlinear mappings with multidimensional input variables. Therefore, an anisotropic kernel with a high-level structure should be constructed, and the training part of GPR should be started by obtaining the training data set and then initializing the hyperparameters. GPR uses the conjugate gradient method to determine the optimal value of the hyperparameters, which will lead to a decrease in the negative marginal log-likelihood function. Finally, RUL is estimated by Equations (6) and (7), which can be expressed as:

$$\mu_* = k_*^T \left(K + \sigma_n^2 I \right)^{-1} y \tag{6}$$

$$\sum_* = \sigma_n^2 + k_{**} - k_*^T \left(K + \sigma_n^2 I \right)^{-1} k_* \tag{7}$$

where μ_* is RUL estimation; $k_* = [k(x_1, x_*), \dots, k(x_N, x_*)]^T$; $k_{**} = k_s(x_*, x_*)$; the kernel matrix is denoted as K, the output of the trained data set is y, I is the identity matrix, and the inverse matrix is determined by the marginal log-likelihood function, and its gradient is $K + \sigma_n^2 I$.
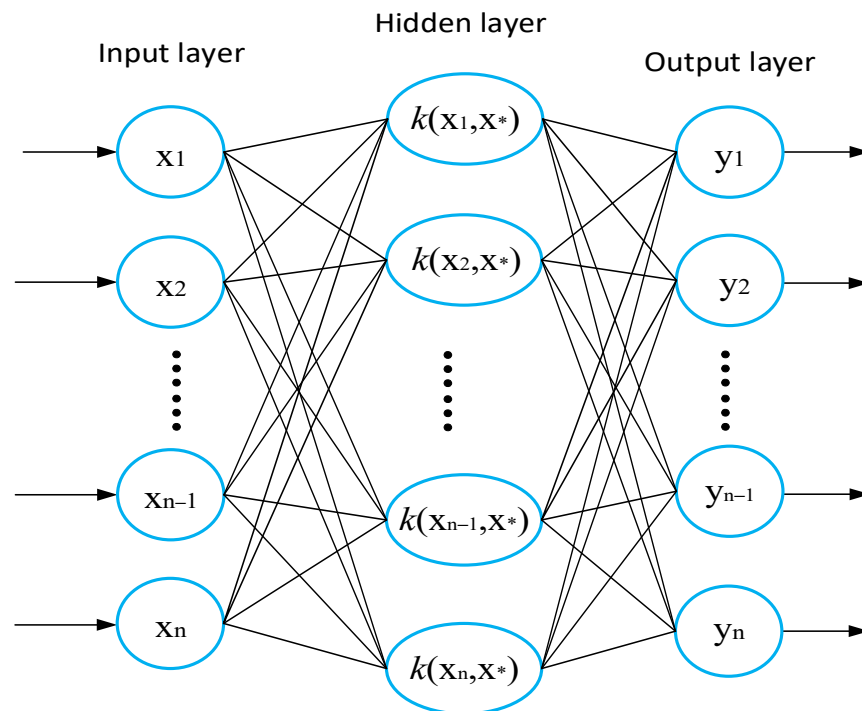

Figure 3. Gaussian process regression algorithm structure diagram.

Although GPR can predict nonlinear systems, as the complexity of prediction increases, the accuracy of GPR will drop rapidly [45]. To solve this problem, GPR introduced a new covariance function and a mean function based on the zero mean function and the diagonal covariance squared exponential function [46,47]. The performance of GPR is highly sensitive to the covariance function, so proper kernel selection and hyperparameter optimization can avoid the problem of excessive sensitivity. To improve ground-penetrating radar, one method is to minimize the negative impact of logarithmic marginal probability [48,49]. It is usually necessary to inverse the covariance matrix to train GPR, which will increase the calculation time and complexity of the algorithm and increase the memory requirements [50]. To solve the problems of long calculation time and high complexity, various sparse methods based on the use of a subset of the training sample size have been developed.

GPR is suitable for processing complex regression problems with high dimensionality, small sample size, and nonlinearity [48]. Yu et al. [51] improved the mixed Gaussian process function regression method, combined with the wavelet denoising data processing method, and used the improved method to predict the RUL of Lithium-ion batteries. The accuracy of this method can reach 2.2%. Compared with the original method, the accuracy is increased by 4.5%. The relative prediction errors of this method are all less than 7%. Li et al. [52] deeply analyzed the changing trend of some incremental capacity and extracted four key feature vectors based on the relationship between capacity change and battery aging. The extracted feature vector will be used as the input data of the Gaussian process regression, and a multi-time scale short-term battery aging model will be constructed using GPR with kernel correction. The mean average error (MAE) and RMSE of this method are both less than 26 cycles. Li et al. [53] combined the characteristics of the equivalent voltage change and the corresponding capacity change with the dual Gaussian process regression model to predict the battery health status. Using this method to estimate the long-term health status of the four batteries, the predicted RUL error is less than 23 cycles.

When selecting the GPR method for RUL prediction, the activation function is usually a kernel function, the training algorithm is a squared exponential kernel or Marginal log-likelihood function, and the hyperparameter adjustment methods are the input dimension length scale and latent function values.

2.3. Extreme Learning Machine

With the development of battery RUL prediction technology, extreme learning machines have also been applied in this field. ELM can randomly select hidden layer unit settings. When the single hidden layer feedforward neural network (SLFN) is determined, ELM can analyze the output weight of SLFN. Because of its fast learning speed and high prediction accuracy, ELM has been widely used in single-step and multi-step prediction algorithms. In the process of state estimation for nonlinear complex systems, ELM has strong flexibility, scalability, and high learning performance, which can quickly approach the real value. ELM is a member of ML, and its structure is usually divided into three layers, namely the input layer, hidden layer, and output layer, as shown in Figure 4.

During the data input process, ELM randomly assigns the input weight. When the data are transmitted between the input layer and the hidden layer, the hidden layer deviation is also set randomly, and the input weight and the hidden layer deviation do not need to be adjusted after setting. When the data pass through the hidden layer and enter the output layer, the connection weight will be determined by solving the equation once. Since the connection weight does not need to be adjusted iteratively, ELM can perform fast convergence. In Figure 4, x_i represents the input layer, and y_i represents the output layer. The mathematical expression output by the hidden layer is represented by the following equation:

$$\sum_{i=1}^{\tilde{N}} \beta_i f_i(x_i) = \sum_{i=1}^{\tilde{N}} \beta_i f\left(a_i x_i + b_j\right) = t_j, j = 1, 2, \ldots, N \tag{8}$$

$$f\left(a_i \cdot x_j + b_j\right) = \frac{1}{1 + e^{-(a_j \cdot x_j + b_j)}}, i = 1, \ldots, L, j = 1, \ldots, N \tag{9}$$

$x = [x_{i1}, x_{i2}, \ldots, x_{iN}]^T$ is the input weight vector, b_i is the hidden layer deviation, $\widetilde{N}$ is the hidden neuron, the weight vector of the *i*-th hidden node and the input node and the output weight of the output layer neuron are denoted as $a_i = [a_{i1}, a_{i2}, \ldots, a_{iN}]^T$ and $B_i = [B_{i1}, \beta_{i2}, \ldots, \beta_{iN}]^T$, respectively. The activation function is $f(x)$ (e.g., sigmoid, tanh, linear, etc.).
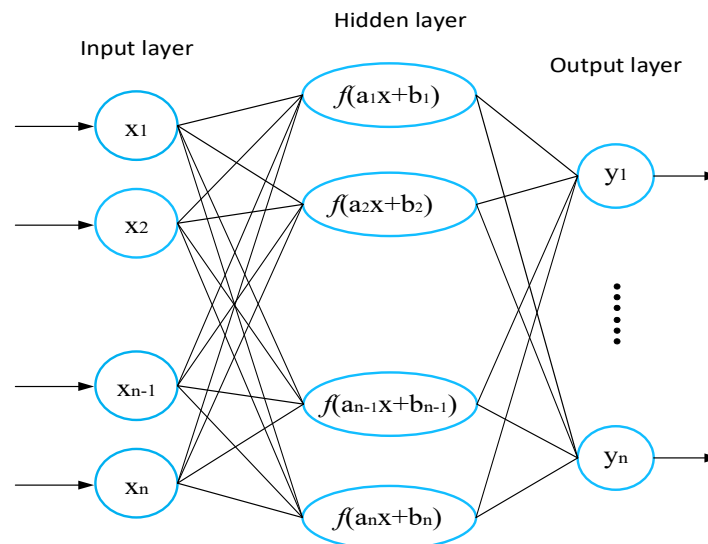

Figure 4. Extreme learning machine algorithm structure diagram.

Zhu et al. [54] developed and optimized the ELM, integrated the gray wolf optimization (GWO) into the ELM algorithm, and improved the weight and threshold of the ELM to form a new DGWO-ELM algorithm. The minimum RMSE of this algorithm can reach 0.43%. Fan et al. [55] also focused on the combination of the hybrid gray wolf optimizer (HGWO) algorithm and ELM and added an attention mechanism to optimize the forgotten online sequential extreme learning machine (FOS-ELM). The RMSE of the improved hybrid method can reach 0.0121.

Guo et al. [56] combined RVFL and ELM to obtain a new hybrid data-driven SOH and RUL joint state estimation model. To quantitatively evaluate the RUL prediction interval, the author developed an uncertainty management method based on bootstrap to improve the accuracy of prediction. Compared with the latest learning algorithm, this method improves the robustness of the model and reduces the prediction error.

When ELM is used as the RUL prediction method, the activation function is usually selected as sigmoid, the training algorithm is a linear system function, and the hyperparameters are adjusted through hidden neurons.

2.4. Deep Neural Network

Unlike the single-layer feedforward neural network (SLFNN) structure of the standard ANN model, the DNN model contains multiple hidden layers. In the DNN algorithm, a functional relationship is established between the input vector and the output vector through nonlinear calculations. In the calculation process, the function parameters are calculated by a certain method. The DNN contains multiple hidden layers, as shown in Figure 5.

$$y = f(x) = f_a\left(W_0^T f_a\left(W_0^T x + b_0\right) + b_1\right) \tag{10}$$

$$f(x) = f_a\left(W_i^T f_a\left(W_{i-1}^T \ldots f_a\left(W_0^T x + b_0\right)\right) + b_i\right) \tag{11}$$

SLFNN is shown in Equation (10), where the input data are x, the output data are y, the activation function is denoted by f_a, and the weight and deviation are W and b, respectively. In the training process, the real value is approached by continuous iterative updating of W. The He method is usually used for initialization. DNN can be described by Equation (11).
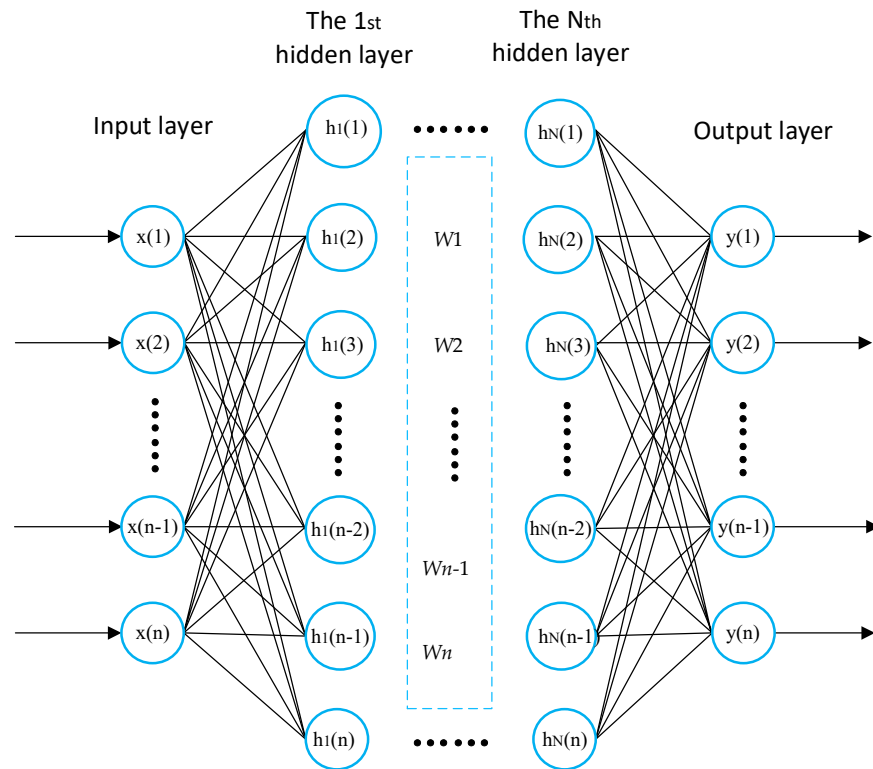

Figure 5. Deep neural network algorithm structure diagram.

In [57], Ma et al. introduced a transfer learning method based on the DNN method. To select the battery with the most similar performance to the target battery as a reference, the average Euclidean distance-based (AED) method with transferable measurement characteristics is used to select in the historical database. Then, the data are used as the input vector to train the prediction model based on the stacked denoising autoencoder (SDA), and finally the RUL of the target battery is obtained. The improved method can increase the prediction speed by nearly 30%. Hong et al. [58] proposed a new DNN prediction model for the long prediction period of Lithium-ion battery RUL. The model uses an end-to-end deep learning framework to achieve the goal of completing RUL predictions through short-term measurements. The average absolute error rate of this method reaches 10.6%. In [59], the author applied DNN to predict the RUL of Lithium-ion batteries in the field of electronic vehicles. The capacity was predicted using 11 extracted features, and two DNNs were trained. One DNN performed statistical analysis on the capacity attenuation of impedance attenuation as the degree of deterioration increased, and the other DNN obtained the probability prediction based on the capacity attenuation trend to improve the predictive accuracy of the remaining service life. The RMSE of this method is approximately 3.59%.

When DNN is used as an RUL prediction method, the activation function is usually selected as sigmoid or ReLU, the training algorithm is gradient descent, backpropagation through time, and the hyperparameters are adjusted through hidden layers.

2.5. Recurrent Neural Network

Recurrent neural networks are widely used to process time-series data because of their time series memory. RNN is an SLFNN, with a classic three-layer model structure.

According to time changes, the time series variables at each moment are used as the input of RNN. Through training, RNN can predict the changing trend of input variables [60]. Among the many RNN architectures, the long and short-term memory (LSTM) algorithm is the most representative [44,61]. LSTM has a forget gate that can filter low-correlation inputs and enhance strong-correlation inputs. In this way, the problem of vanishing and exploding gradients can be solved [62]. Compared with the traditional RNN algorithm, LSTM is more suitable for scenarios that require long-term prediction and has better robustness and accuracy [63].

Figure 6 shows the typical structure of an RNN. For an input sample $x = [x_1, x_2, \ldots, x_{N,}]$ where N denotes the sequence length, RNN calculates the hidden state vector sequence $h = [h_1, h_2, \ldots, h_{N,}]$ and outputs the sequence $y = [y_1, y_2, \ldots, y_{N,}]$ through iteration of the Equations (12) and (13), from $t = 1$ to N.

$$h_t = \mathcal{H}(W_{xh}x_t + W_{hh}h_{t-1} + b_h) \tag{12}$$

$$y_t = W_{hy}h_t + b_y \tag{13}$$

where the weight and bias are W and b, respectively. The weight before the input layer and the hidden layer is represented by W_{xh}, and the bias vector and the nonlinear activation function of the hidden layer are b_h and $\mathcal{H}$, respectively.
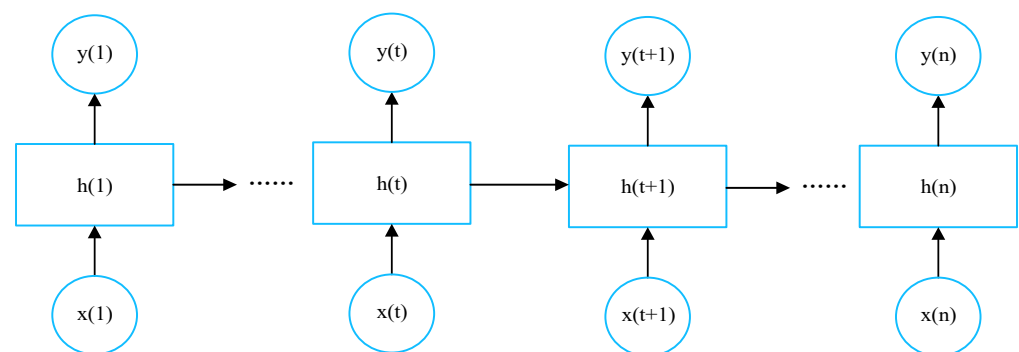

Figure 6. Recurrent neural network algorithm structure diagram.

Wu et al. [19] applied the bat particle filter (Bat-PF) to optimize the neural network algorithm. The formed NN+Bat-PF model uses Bat-PF to recursively update the model parameters. The error of predicting RUL is two cycles in 500 prediction cycles, and the width of the probability density function (PDF) is 35 cycles. She et al. [7], based on the radial basis function NN model, used the incremental capacity analysis method to analyze the battery capacity aging trend, and the RUL was predicted based on the relationship between the capacity and the remaining service lifetime of the battery. The prediction accuracy of this method and MAE are 90% and 4.00%, respectively. To realize the online estimation of the RUL of Lithium-ion batteries, Wu et al. [64] used the importance sampling (IS) method to process historical data sets. The feature vector is selected as the input of the feedforward neural network (FFNN), and 40 hidden layer neurons are used for training. This improved online estimation method has an error of less than 5% in actual operation. For online RUL estimation, Zhang et al. [65] combined the incremental capacity analysis method while simplifying the ANN model. There are only two neurons in the input layer of the simplified ANN model. The maximum MAE of this method is four cycles, and the maximum RMSE is six cycles.

Based on the RNN structure, an LSTM architecture is used. The RNN algorithm uses the backpropagation method for training, but this method usually brings about the problem of gradient explosion or disappearance. LSTM uses memory cells instead of hidden nodes to solve this problem. Figure 7 shows the structure of a single LSTM memory cell. At each time step, the storage unit is accessed, updated, and cleared by multiple gates. The input vector of the LSTM unit at time t is x_t, the hidden state is expressed as h_t, c_t is the unit

memory, the weight matrix and bias parameters are W and b, respectively, the activation function of the input gate is i_t, the activation function of the forgetting gate is f_t, and the activation function of the output gate is o_t. When new input data are fed into the cell, the information is accumulated to the memory cell if the input gate i_t is activated. The previous cell state c_{t-1} can be "forgotten" if the forget gate f_t is on. The output gate o_t determines whether the newest cell output c_t can be propagated to the final status h_t, and $\mathcal{H}$ is implemented as:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \tag{14}$$

$$f_t = \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f\right) \tag{15}$$

$$c_t = f_t c_{t-1} + i_t tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \tag{16}$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o) \tag{17}$$

$$h_t = o_t tanh(c_t) \tag{18}$$

where σ represents the logistic sigmoid function, W_{hi} denotes the hidden-input gate matrix, and W_{xo} is the input/output gate matrix. The LSTM algorithm structure is shown in Figure 7.
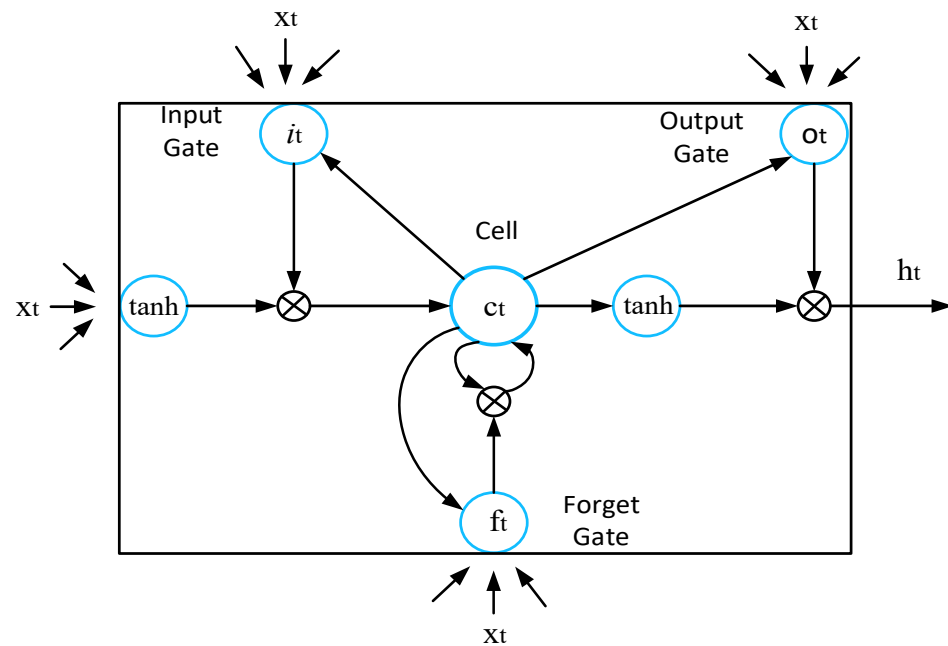

Figure 7. LSTM algorithm structure diagram.

Li et al. [66] proposed an Elman-LSTM method. This method combines the time memory of LSTM and the advantages of the Elman neural network and introduces the empirical mode decomposition algorithm into it. The relative prediction errors of this Elman-LSTM method are 3.3% and 3.21%, respectively. Qu et al. [67] combined a simple and easy-to-implement particle swarm optimization algorithm with LSTM training and further introduced an attention mechanism to achieve the effect of joint state estimation of SOH and RUL. The average error of this method is -3 and the RMSE is 0.0362. Cui et al. [68] introduced the unscented Kalman filter (UKF) algorithm based on the neural network framework of LSTM and NN, forming a new data-driven hybrid model method. The average error of this method is 5. Yang et al. [69] combined the optimized bidirectional long short-term memory network (Bi-LSTM) with the convolutional neural network, which is the same neural network algorithm. The minimum error of this hybrid neural network algorithm is 1.04%. Chinomona et al. [70] proposed a forward feature selection algorithm

that uses a combination of RNN and LSTM to completely select the best feature set. Using partial charge/discharge data, the RMSE and MAE of this method are 0.00286 and 0.00222, respectively.

Ma et al. [71] combined the convolutional neural network with the LSTM method and merged the resulting hybrid method with the false nearest neighbor (FNN) method. The accuracy of this method is 98.21%. Qiao et al. [72] combined the empirical mode decomposition method suitable for processing nonlinear non-stationary signals with DNN with nonlinear system prediction advantages and LSTM with temporal memory characteristics to predict RUL. Compared with traditional methods, the algorithm's MAE and RMSE, which are 75% and 90.8%, respectively, significantly decrease. The standard deviation of this method is 1.36626. Li et al. [73] designed a variant of LSTM called AST-LSTM NN. AST-LSTM NN has many-to-one and one-to-one mapping structures. This method predicts that the absolute error of RUL is 0.0831. Liu et al. [74] combined the advantages of LSTM and GPR. LSTM can accurately predict the long-term dynamic trend of capacity degradation, and the prediction deviation caused by capacity regeneration can be accurately captured by GPR. The RMSE and maximum error of the LSTM+GPR model are 0.0032 and 0.6%, respectively. Parker et al. [75] proposed a many-to-one framework based on LSTM to adapt to various input types. The mean absolute percentage error (MAPE) of the proposed model is 63.7% higher than that of the traditional method.

When RNN performs RUL prediction, the activation function is usually selected as sigmoid, the training algorithm is gradient descent, backpropagation through time, and potential overfitting problems are solved through hyperparameter adjustment. Among them, the activation function of the LSTM algorithm is usually selected as sigmoid and tanh, the training algorithm is gradient descent, backpropagation through time, and the hyperparameters are adjusted through hidden neurons.

3. Comparison

Even if the working conditions remain the same, Lithium-ion batteries will not necessarily show a linear degradation behavior (e.g., capacity fade, resistance increase, power decrease, etc.) during their life. Therefore, the ideal RUL prediction method should be able to consider these nonlinear behaviors. If the prediction method only focuses on minimizing the error, it may lead to the problem of overfitting. The accuracy of data-driven methods depends on the correct adjustment of hyperparameters. The training data can contain valuable measurement noise indicators. Therefore, the forecasting method should consider uncertain factors. The performance of various RUL prediction methods can be evaluated from the following aspects: (1) activation function; (2) training algorithm; (3) hyperparameter adjustment; (4) uncertainty management; (5) robustness. These aspects are shown in Table 2.

Choosing an appropriate amount of data is essential to obtain a satisfactory RUL estimation result. In actual operation, online learning is more practical. In this case, the scale of training vectors gradually increases over time, and a large number of data sets may cause a huge computational burden. Considering the limited memory and computing power, it is necessary to know the complexity of the input and output vectors and the algorithm structure of each method. The accuracy of the RUL estimation greatly depends on collecting the relevant data. Normally, the original data will be normalized to shorten the training time and improve the performance of the algorithm. The following factors can be considered to evaluate the performance of various RUL prediction methods: (1) input features and output; (2) structure; (3) data calculation. These factors are shown in Table 3.

Table 2. Summaries of the different criteria used for accuracy assessment defined in this section.

Method	Activation Function		Training Algorithm		Hyperparameter Adjustment		Uncertainty Management		Robustness
SVM	-	Radial based kernel	-	Logistic regression and functional margin	- - -	Regularization factor SVM type regression Kernel parameter	-	Not probabilistic	-	High robustness against small deviations
GPR	-	Kernel function	- -	Squared exponential kernel Marginal loglikelihood function	- -	Input dimension length scale Latent function values	-	Probabilistic	-	Related to the selected kernel
ELM	-	Sigmoid	-	Linear system function	-	Hidden neurons	-	Not probabilistic	-	Appropriate value of hidden neurons
DNN	- -	Sigmoid ReLU	- -	Gradient descent Backpropagation through time	-	Hidden layers	-	Not probabilistic	-	Depends on the number of hidden layers
RNN	-	Sigmoid	- -	Gradient descent Backpropagation through time	-	Potential overfitting problems	-	Not probabilistic	-	Potential overfitting problems
LSTM	- -	Sigmoid tanh	- -	Gradient descent Backpropagation through time	-	Hidden neurons	-	Not probabilistic	-	Stable long-term forecast

Table 3. Summaries of the different criteria for computational complexity evaluation defined in this section.

Method	Input Features and Output		Structure		Data Calculation
SVM	- -	Feature vector = [I (t), V (t), T (t), SOC (t), Cap (t)] Output = [RUL (t)]	-	Radial-based kernel function	-	Exponential function, multiply and accumulate
GPR	- -	Feature vector = [I (t), V (t), Cap (t)] Output = [RUL (t)]	-	Single hidden layer	-	Exponential function, multiply and accumulate
ELM	- -	Feature vector = [I (t), V(t), T (t), R (t), PR (t), Cap (t)] Output = [RUL (t)]	-	Single hidden layer	-	Exponential function, multiply and accumulate
DNN	- -	Feature vector = [I(t), V (t), T (t), Cap (t)] Output = [RUL (t)]	-	Multiple hidden layers	-	Exponential functions, vector and matrix operations
RNN	- -	Feature vector = [I (t), T (t), SOC (t), R/C (t), Cap (t)] Output = [RUL (t)]	- -	Single hidden layer Prediction window	-	Exponential functions, vector and matrix operations
LSTM	- -	Feature vector = [I (t), V (t), Cap (t)] Output = [RUL (t)]	-	Prediction window	-	Exponential functions, vector and matrix operations

Based on the previous summary, the advantages and disadvantages of the proposed methods are compared, as shown in Table 4. Faced with computational challenges, sparsity may become a key function to solve the problem of excessive input data. SVR becomes a sparse algorithm due to its sensitive loss function. SVM has satisfactory performance in nonlinear and high-dimensional models, can deal with local minima and small sample sizes, and has a short calculation time. However, it cannot express uncertainty due to its difficulty with calculating kernel and regularization parameters. GPR is not a sparse model, but different data processing methods can be used to reduce the training data size. Due to the non-parametric nature and execution probability of the GPR method, it has better robustness and computational efficiency prediction capabilities. Since the covariance

provided by GPR shows excellent uncertainty management capabilities, it has strong flexibility and adaptability when dealing with high-dimensional and small sample data sets. However, when it is applied to high-dimensional space, the efficiency is reduced, the kernel function seriously affects the performance, and the amount of calculation is large.

ELM has better scalability and generalization performance, simple structure, and low computational complexity. Furthermore, its accuracy is determined by the value of the hidden neuron. DNN has a strong independent learning ability and generalization ability and high algorithm accuracy, suitable for nonlinear and complex systems. Its performance depends on the number of hidden layer neurons and the number of input historical data; it needs enough training data, the structure is complex, and the memory consumption is large. RNN has high prediction accuracy, is suitable for nonlinear and complex systems, and has strong long-term RUL prediction capabilities. However, its uncertainty management ability is poor and there is a problem of overfitting. LSTM has satisfactory results under long-term dependence, and the computational intensity of the online phase is low. However, this method has a lengthy and complicated training process and requires expensive equipment to accelerate training.

Table 4. Advantages and disadvantages of the different ML techniques used for RUL prediction.

Method	Advantages	Disadvantages
SVM	- Satisfactory performance in non-linear and high-dimension models. - Capable of dealing with the local minimum, non-linear, and small sample size problems - The global optimal solution can be obtained - High prediction accuracy - Less prediction times - Non-parametric	- Kernel and regularization parameters are difficult to calculate - Low expression of uncertainty - Kernel functions need to satisfy the Mercer criterion
GPR	- Provides covariance to generate uncertainty level - Easy interpretability of features - Processes high-dimensional and small sample data sets - Strong flexibility and adaptability - Non-parametric	- Performance is highly affected by kernel functions - Low efficiency in high dimensional spaces - High computational load with large data sets - Hyperparameter adjustment is complex - Lack of sparseness
ELM	- High scalability - Strong generalization performance - Low computational complexity - Fast learning	- The value of the hidden neuron determines the accuracy
DNN	- Strong independent learning ability - Suitable for non-linear and complex systems - Better generalization capability - High algorithm accuracy	- The number of hidden layers and past inputs determines performance - Requires sufficient training data - Complex structure - Large memory consumption
RNN	- High prediction accuracy - Suitable for non-linear and complex systems - Strong long-term RUL prediction ability	- High risk of overfitting - Lack of uncertainty management capabilities - Accuracy depends on the training process - Requires expensive processing unit - Large memory consumption
LSTM	- Has satisfactory outcomes under long-term dependencies - Low sensitivity of online calculation	- Long and complex training execution - High cost of accelerated training

4. Challenges and Prospects

ML is the preferred method of using historical data sets generated by cycles to predict future development trends. Among them, DNN has a strong independent learning ability and generalization ability, which makes DNN more suitable for RUL prediction. Due to the excellent adaptability of ML, it is suitable for strongly non-linear systems and fits the true trajectory of the system by automatically optimizing model parameters. However, its accuracy relies on a large amount of historical data inputs to train the algorithm, which is

also its inevitable limitation. In actual operation, a large amount of training data inputs will increase the calculation time and computational complexity, and it is also easy to cause data overfitting. There needs to be a balance between using ML algorithms to improve the accuracy of prediction and computational complexity.

With the emergence of more and more battery state estimation methods, combined with the application of actual operating systems, online state estimation methods will become the trend of future development. BMS will also be upgraded from a traditional offline system to an online management system. In terms of the types of battery state estimation methods, single state estimation will also be upgraded to joint state estimation. In actual operation, there is a coupling relationship between the battery states, and the joint state estimation has better practicability and higher accuracy. It can be expected that multi-state collaborative real-time online management solutions based on artificial intelligence will become the future development direction.

ML algorithms are consistent with the latest developments in artificial intelligence. The future direction of data-driven Lithium-ion battery RUL prediction will focus on developing hybrid ML models that are widely applicable to multiple types of prediction data. Real-time online ML battery management solutions based on big data and cloud computing platforms are expected to become the main method for future Lithium-ion battery RUL predictions.

5. Conclusions

This paper reviews the ML-based RUL prediction methods for Lithium-ion batteries, which are proposed in the literature. An innovative standard is defined to evaluate the accuracy and computational cost of the RUL prediction method. From the above comparison, from the perspective of computational complexity, SVM, GPR, and ELM have the characteristics of simple structure and small calculation amount, but they are more suitable for calculation problems with small sample sizes. From the perspective of prediction accuracy, DNN, RNN, and LSTM all have good performance and are suitable for nonlinear complex systems, such as Lithium-ion batteries. Among them, RNN has a relatively poor ability of uncertainty management, and LSTM has a long and complicated training process and requires expensive equipment to accelerate training. In summary, DNN has a strong independent learning ability and generalization ability, making DNN more suitable for RUL prediction.

References

1. Gao, Y.; Zhang, X.; Guo, B.; Zhu, C.; Wiedemann, J.; Wang, L.; Cao, J. Health-Aware Multiobjective Optimal Charging Strategy with Coupled Electrochemical-Thermal-Aging Model for Lithium-Ion Battery. *IEEE Trans. Ind. Inform.* **2019**, *16*, 3417–3429. [CrossRef]
2. Peng, J.; Zhou, Z.; Wang, J.; Wu, D.; Guo, Y. Residual remaining useful life prediction method for lithium-ion batteries in satellite with incomplete healthy historical data. *IEEE Access* **2019**, *7*, 127788–127799. [CrossRef]
3. Gabbar, H.A.; Othman, A.M.; Abdussami, M.R. Review of Battery Management Systems (BMS) Development and Industrial Standards. *Technologies* **2021**, *9*, 28. [CrossRef]
4. Xing, Y.; Ma, E.W.M.; Tsui, K.L.; Pecht, M. Battery management systems in electric and hybrid vehicles. *Energies* **2011**, *4*, 1840–1857. [CrossRef]
5. Sun, J.; Pei, L.; Liu, R.; Ma, Q.; Tang, C.; Wang, T. Economic Operation Optimization for 2nd Use Batteries in Battery Energy Storage Systems. *IEEE Access* **2019**, *7*, 41852–41859. [CrossRef]
6. Liu, K.; Hu, X.; Wei, Z.; Li, Y.; Jiang, Y. Modified Gaussian Process Regression Models for Cyclic Capacity Prediction of Lithium-Ion Batteries. *IEEE Trans. Transp. Electrif.* **2019**, *5*, 1225–1236. [CrossRef]

7. She, C.; Wang, Z.; Sun, F.; Liu, P.; Zhang, L. Battery Aging Assessment for Real-World Electric Buses Based on Incremental Capacity Analysis and Radial Basis Function Neural Network. *IEEE Trans. Ind. Inform.* **2020**, *16*, 3345–3354. [CrossRef]

8. Xu, B.; Zhao, J.; Zheng, T.; Litvinov, E.; Kirschen, D.S. Factoring the Cycle Aging Cost of Batteries Participating in Electricity Markets. *IEEE Trans. Power Syst.* **2017**, *33*, 2248–2259. [CrossRef]

9. Qi, N.; Dai, K.; Yi, F.; Wang, X.; You, Z.; Zhao, J. An Adaptive Energy Management Strategy to Extend Battery Lifetime of Solar Powered Wireless Sensor Nodes. *IEEE Access* **2019**, *7*, 88289–88300. [CrossRef]

10. Zhang, C.; Wang, Y.; Gao, Y.; Wang, F.; Mu, B.; Zhang, W. Accelerated fading recognition for lithium-ion batteries with Nickel-Cobalt-Manganese cathode using quantile regression method. *Appl. Energy* **2019**, *256*, 113841. [CrossRef]

11. Liu, H.; Chen, F.; Tong, Y.; Wang, Z.; Yu, X.; Huang, R. Impacts of Driving Conditions on EV Battery Pack Life Cycle. *World Electr. Veh. J.* **2020**, *11*, 17. [CrossRef]

12. Corno, M.; Pozzato, G. Active adaptive battery aging management for electric vehicles. *IEEE Trans. Veh. Technol.* **2019**, *69*, 258–269. [CrossRef]

13. Liu, K.; Li, Y.; Hu, X.; Lucu, M.; Widanage, W.D. Gaussian Process Regression with Automatic Relevance Determination Kernel for Calendar Aging Prediction of Lithium-Ion Batteries. *IEEE Trans. Ind. Inform.* **2019**, *16*, 3767–3777. [CrossRef]

14. Ma, J.; Xu, S.; Shang, P.; Ding, Y.; Qin, W.; Cheng, Y.; Lu, C.; Su, Y.; Chong, J.; Jin, H.; et al. Cycle life test optimization for different Li-ion power battery formulations using a hybrid remaining-useful-life prediction method. *Appl. Energy* **2020**, *262*, 114490. [CrossRef]

15. Jamshidi, M.; Jamshidi, M.; Rostami, S. An intelligent approach for nonlinear system identification of a Li-ion battery. In Proceedings of the 2017 IEEE 2nd International Conference on Automatic Control and Intelligent Systems (I2CACIS), Kota Kinabalu, Malaysia, 21 October 2021; pp. 98–103. [CrossRef]

16. Jamshidi, M.B.; Alibeigi, N.; Lalbakhsh, A.; Roshani, S. An ANFIS approach to modeling a small satellite power source of NASA. In Proceedings of the 2019 IEEE 16th International Conference on Networking, Sensing and Control (ICNSC), Banff, AB, Canada, 9–11 May 2019; pp. 459–464. [CrossRef]

17. Shi, Y.; Xu, B.; Tan, Y.; Kirschen, D.; Zhang, B. Optimal Battery Control under Cycle Aging Mechanisms in Pay for Performance Settings. *IEEE Trans. Automat. Contr.* **2019**, *64*, 2324–2339. [CrossRef]

18. Behdad Jamshidi, M.; Rostami, S. A dynamic artificial neural network approach to estimate thermal behaviors of li-ion batteries. In Proceedings of the 2017 IEEE 2nd International Conference on Automatic Control and Intelligent Systems (I2CACIS), Kota Kinabalu, Malaysia, 21 October 2021; pp. 116–121. [CrossRef]

19. Wu, Y.; Li, W.; Wang, Y.; Zhang, K. Remaining useful life prediction of lithium-ion batteries using neural network and bat-based particle filter. *IEEE Access* **2019**, *7*, 54843–54854. [CrossRef]

20. El Mejdoubi, A.; Chaoui, H.; Gualous, H.; Van Den Bossche, P.; Omar, N.; Van Mierlo, J. Lithium-ion batteries health prognosis considering aging conditions. *IEEE Trans. Power Electron.* **2019**, *34*, 6834–6844. [CrossRef]

21. Xue, Z.; Zhang, Y.; Cheng, C.; Ma, G. Remaining useful life prediction of lithium-ion batteries with adaptive unscented kalman filter and optimized support vector regression. *Neurocomputing* **2020**, *376*, 95–102. [CrossRef]

22. Xiong, R.; Li, L.; Tian, J. Towards a smarter battery management system: A critical review on battery state of health monitoring methods. *J. Power Sources* **2018**, *405*, 18–29. [CrossRef]

23. Hu, X.; Xu, L.; Lin, X.; Pecht, M. Battery Lifetime Prognostics. *Joule* **2020**, *4*, 310–346. [CrossRef]

24. Guha, A.; Patra, A. Online Estimation of the Electrochemical Impedance Spectrum and Remaining Useful Life of Lithium-Ion Batteries. *IEEE Trans. Instrum. Meas.* **2018**, *67*, 1836–1849. [CrossRef]

25. Wang, Y.; Tian, J.; Sun, Z.; Wang, L.; Xu, R.; Li, M.; Chen, Z. A comprehensive review of battery modeling and state estimation approaches for advanced battery management systems. *Renew. Sustain. Energy Rev.* **2020**, *131*, 110015. [CrossRef]

26. Xiong, R.; Zhang, Y.; Wang, J.; He, H.; Peng, S.; Pecht, M. Lithium-Ion Battery Health Prognosis Based on a Real Battery Management System Used in Electric Vehicles. *IEEE Trans. Veh. Technol.* **2018**, *68*, 4110–4121. [CrossRef]

27. Motaqi, A.; Mosavi, M.R. Blind and task-ware multi-cell battery management system. *Eng. Sci. Technol. Int. J.* **2020**, *23*, 544–554. [CrossRef]

28. Chen, W.; Liang, J.; Yang, Z.; Li, G. A review of lithium-ion battery for electric vehicle applications and beyond. *Energy Procedia* **2019**, *158*, 4363–4368. [CrossRef]

29. Tian, H.; Qin, P.; Li, K.; Zhao, Z. A review of the state of health for lithium-ion batteries: Research status and suggestions. *J. Clean. Prod.* **2020**, *261*, 120813. [CrossRef]

30. Lin, C.P.; Cabrera, J.; Yang, F.; Ling, M.H.; Tsui, K.L.; Bae, S.J. Battery state of health modeling and remaining useful life prediction through time series model. *Appl. Energy* **2020**, *275*, 115338. [CrossRef]

31. Lipu, M.S.H.; Hannan, M.A.; Hussain, A.; Hoque, M.M.; Ker, P.J.; Saad, M.H.M.; Ayob, A. A review of state of health and remaining useful life estimation methods for lithium-ion battery in electric vehicles: Challenges and recommendations. *J. Clean. Prod.* **2018**, *205*, 115–133. [CrossRef]

32. Hui, Y.; Li, M.; Downey, A.; Shen, S.; Pavan, V.; Ye, H.; Vanelzen, C.; Jain, G.; Hu, S.; Laflamme, S.; et al. Physics-based prognostics of implantable-grade lithium-ion battery for remaining useful life prediction. *J. Power Sources* **2021**, *485*, 229327. [CrossRef]

33. Khumprom, P.; Yodo, N. A data-driven predictive prognostic model for lithium-ion batteries based on a deep learning algorithm. *Energies* **2019**, *12*, 660. [CrossRef]

34. Liu, Z.; Sun, G.; Bu, S.; Han, J.; Tang, X.; Pecht, M. Particle Learning Framework for Estimating the Remaining Useful Life of Lithium-Ion Batteries. *IEEE Trans. Instrum. Meas.* **2017**, *66*, 280–293. [CrossRef]

35. Xiong, R.; Lu, J. Preparation of Papers for IFAC Conferences & Symposia: A comparative study of remaining useful life predictions for lithium-ion battery. *IFAC-PapersOnLine* **2018**, *51*, 268–273. [CrossRef]

36. Lucu, M.; Martinez-Laserna, E.; Gandiaga, I.; Camblong, H. A critical review on self-adaptive Li-ion battery ageing models. *J. Power Sources* **2018**, *401*, 85–101. [CrossRef]

37. Gao, T.; Lu, W. Machine learning toward advanced energy storage devices and systems. *iScience* **2021**, *24*, 101936. [CrossRef] [PubMed]

38. Hossain Lipu, M.S.; Hannan, M.A.; Karim, T.F.; Hussain, A.; Saad, M.H.M.; Ayob, A.; Miah, M.S.; Indra Mahlia, T.M. Intelligent algorithms and control strategies for battery management system in electric vehicles: Progress, challenges and future outlook. *J. Clean. Prod.* **2021**, *292*, 126044. [CrossRef]

39. Li, Y.; Liu, K.; Foley, A.M.; Zülke, A.; Berecibar, M.; Nanini-Maury, E.; Van Mierlo, J.; Hoster, H.E. Data-driven health estimation and lifetime prediction of lithium-ion batteries: A review. *Renew. Sustain. Energy Rev.* **2019**, *113*, 109254. [CrossRef]

40. Zhang, H.; Mo, Z.; Wang, J.; Miao, Q. Nonlinear-drifted fractional brownian motion with multiple hidden state variables for remaining useful life prediction of lithium-ion batteries. *IEEE Trans. Reliab.* **2020**, *69*, 768–780. [CrossRef]

41. Patil, M.A.; Tagade, P.; Hariharan, K.S.; Kolake, S.M.; Song, T.; Yeo, T.; Doo, S. A novel multistage Support Vector Machine based approach for Li ion battery remaining useful life estimation. *Appl. Energy* **2015**, *159*, 285–297. [CrossRef]

42. Zhao, Q.; Qin, X.; Zhao, H.; Feng, W. A novel prediction method based on the support vector regression for the remaining useful life of lithium-ion batteries. *Microelectron. Reliab.* **2018**, *85*, 99–108. [CrossRef]

43. Du, J.; Zhang, W.; Zhang, C.; Zhou, X. Battery remaining useful life prediction under coupling stress based on support vector regression. *Energy Procedia* **2018**, *152*, 538–543. [CrossRef]

44. Hossain Lipu, M.S.; Hannan, M.A.; Hussain, A.; Ayob, A.; Saad, M.H.M.; Karim, T.F.; How, D.N.T. Data-driven state of charge estimation of lithium-ion batteries: Algorithms, implementation factors, limitations and future trends. *J. Clean. Prod.* **2020**, *277*, 124110. [CrossRef]

45. Liu, J.; Chen, Z. Remaining useful life prediction of lithium-ion batteries based on health indicator and Gaussian process regression model. *IEEE Access* **2019**, *7*, 39474–39484. [CrossRef]

46. Cong, X.; Zhang, C.; Jiang, J.; Zhang, W.; Jiang, Y. A Hybrid Method for the Prediction of the Remaining Useful Life of Lithium-Ion Batteries with Accelerated Capacity Degradation. *IEEE Trans. Veh. Technol.* **2020**, *69*, 12775–12785. [CrossRef]

47. Hu, X.; Che, Y.; Lin, X.; Deng, Z. Health Prognosis for Electric Vehicle Battery Packs: A Data-Driven Approach. *IEEE/ASME Trans. Mechatronics* **2020**, *25*, 2622–2632. [CrossRef]

48. Johnen, M.; Schmitz, C.; Kateri, M.; Kamps, U. Fitting lifetime distributions to interval censored cyclic-aging data of lithium-ion batteries. *Comput. Ind. Eng.* **2020**, *143*, 106418. [CrossRef]

49. Kang, W.; Xiao, J.; Xiao, M.; Hu, Y.; Zhu, H.; Li, J. Research on Remaining Useful Life Prognostics Based on Fuzzy Evaluation-Gaussian Process Regression Method. *IEEE Access* **2020**, *8*, 71965–71973. [CrossRef]

50. Yu, J. State of health prediction of lithium-ion batteries: Multiscale logic regression and Gaussian process regression ensemble. *Reliab. Eng. Syst. Saf.* **2018**, *174*, 82–95. [CrossRef]

51. Yu, P.; Hou, Y.; Song, Y.; Pang, J.; Liu, D. Lithium-Ion Battery Prognostics with Hybrid Gaussian Process Function Regression. *Energies* **2018**, *11*, 1420. [CrossRef]

52. Li, X.; Yuan, C.; Wang, Z. Multi-time-scale framework for prognostic health condition of lithium battery using modified Gaussian process regression and nonlinear regression. *J. Power Sources* **2020**, *467*, 228358. [CrossRef]

53. Li, X.; Wang, Z.; Yan, J. Prognostic health condition for lithium battery using the partial incremental capacity and Gaussian process regression. *J. Power Sources* **2019**, *421*, 56–67. [CrossRef]

54. Zhu, J.; Tan, T.; Wu, L.; Yuan, H. RUL prediction of lithium-ion battery based on improved DGWO-ELM method in a random discharge rates environment. *IEEE Access* **2019**, *7*, 125176–125187. [CrossRef]

55. Fan, J.; Fan, J.; Liu, F.; Qu, J.; Li, R. A Novel Machine Learning Method Based Approach for Li-Ion Battery Prognostic and Health Management. *IEEE Access* **2019**, *7*, 160043–160061. [CrossRef]

56. Gou, B.; Xu, Y.; Feng, X. State-of-Health Estimation and Remaining-Useful-Life Prediction for Lithium-Ion Battery Using a Hybrid Data-Driven Method. *IEEE Trans. Veh. Technol.* **2020**, *69*, 10854–10867. [CrossRef]

57. Ma, J. Remaining Useful Life Transfer Prediction and Cycle Life Test Optimization for Different Formula Li-ion Power Batteries Using a Robust Deep Learning Method. *IFAC Pap.* **2020**, *53*, 54–59. [CrossRef]

58. Hong, J.; Lee, D.; Jeong, E.R.; Yi, Y. Towards the swift prediction of the remaining useful life of lithium-ion batteries with end-to-end deep learning. *Appl. Energy* **2020**, *278*, 115646. [CrossRef]

59. Lee, C.J.; Kim, B.K.; Kwon, M.K.; Nam, K.; Kang, S.W. Real-time prediction of capacity fade and remaining useful life of lithium-ion batteries based on charge/discharge characteristics. *Electronics* **2021**, *10*, 846. [CrossRef]

60. Tang, X.; Liu, K.; Wang, X.; Gao, F.; MacRo, J.; Widanage, W.D. Model Migration Neural Network for Predicting Battery Aging Trajectories. *IEEE Trans. Transp. Electrif.* **2020**, *6*, 363–374. [CrossRef]

61. Zhang, Y.; Xiong, R.; He, H.; Pecht, M.G. Long short-term memory recurrent neural network for remaining useful life prediction of lithium-ion batteries. *IEEE Trans. Veh. Technol.* **2018**, *67*, 5695–5705. [CrossRef]

62. Li, W.; Jiao, Z.; Du, L.; Fan, W.; Zhu, Y. An indirect RUL prognosis for lithium-ion battery under vibration stress using Elman neural network. *Int. J. Hydrogen Energy* **2019**, *44*, 12270–12276. [CrossRef]

63. Zhou, D.; Li, Z.; Zhu, J.; Zhang, H.; Hou, L. State of Health Monitoring and Remaining Useful Life Prediction of Lithium-Ion Batteries Based on Temporal Convolutional Network. *IEEE Access* **2020**, *8*, 53307–53320. [CrossRef]

64. Wu, J.; Zhang, C.; Chen, Z. An online method for lithium-ion battery remaining useful life estimation using importance sampling and neural networks. *Appl. Energy* **2016**, *173*, 134–140. [CrossRef]

65. Zhang, S.; Zhai, B.; Guo, X.; Wang, K.; Peng, N.; Zhang, X. Synchronous estimation of state of health and remaining useful lifetime for lithium-ion battery using the incremental capacity and artificial neural networks. *J. Energy Storage* **2019**, *26*, 100951. [CrossRef]

66. Li, X.; Zhang, L.; Wang, Z.; Dong, P. Remaining useful life prediction for lithium-ion batteries based on a hybrid model combining the long short-term memory and Elman neural networks. *J. Energy Storage* **2019**, *21*, 510–518. [CrossRef]

67. Qu, J.; Liu, F.; Ma, Y.; Fan, J. A Neural-Network-Based Method for RUL Prediction and SOH Monitoring of Lithium-Ion Battery. *IEEE Access* **2019**, *7*, 87178–87191. [CrossRef]

68. Cui, X.; Hu, T. State of Health Diagnosis and Remaining Useful Life Prediction for Lithium-ion Battery Based on Data Model Fusion Method. *IEEE Access* **2020**, *8*, 207298–207307. [CrossRef]

69. Yang, H.; Wang, P.; An, Y.; Shi, C.; Sun, X.; Wang, K.; Zhang, X.; Wei, T.; Ma, Y. Remaining useful life prediction based on denoising technique and deep neural network for lithium-ion capacitors. *eTransportation* **2020**, *5*, 100078. [CrossRef]

70. Chinomona, B.; Chung, C.; Chang, L.-K.; Su, W.-C.; Tsai, M.-C. Long Short-Term Memory Approach to Estimate Battery Remaining Useful Life Using Partial Data. *IEEE Access* **2020**, *8*, 165419–165431. [CrossRef]

71. Ma, G.; Zhang, Y.; Cheng, C.; Zhou, B.; Hu, P.; Yuan, Y. Remaining useful life prediction of lithium-ion batteries based on false nearest neighbors and a hybrid neural network. *Appl. Energy* **2019**, *253*, 113626. [CrossRef]

72. Qiao, J.; Liu, X.; Chen, Z. Prediction of the Remaining Useful Life of Lithium-Ion Batteries Based on Empirical Mode Decomposition and Deep Neural Networks. *IEEE Access* **2020**, *8*, 42760–42767. [CrossRef]

73. Li, P.; Zhang, Z.; Xiong, Q.; Ding, B.; Hou, J.; Luo, D.; Rong, Y.; Li, S. State-of-health estimation and remaining useful life prediction for the lithium-ion battery based on a variant long short term memory neural network. *J. Power Sources* **2020**, *459*, 228069. [CrossRef]

74. Liu, K.; Shang, Y.; Ouyang, Q.; Widanage, W.D. A Data-driven Approach with Uncertainty Quantification for Predicting Future Capacities and Remaining Useful Life of Lithium-ion Battery. *IEEE Trans. Ind. Electron.* **2020**, *68*, 3170–3180. [CrossRef]

75. Park, K.; Choi, Y.; Choi, W.J.; Ryu, H.Y.; Kim, H. LSTM-Based Battery Remaining Useful Life Prediction with Multi-Channel Charging Profiles. *IEEE Access* **2020**, *8*, 20786–20798. [CrossRef]